CISCO SYSTEMS

Catalyst 6500 シリーズ スイッチ コンテント スイッチング モジュール コンフィギュレーション ノート

Software Release 4.2(1)

Text Part Number: OL-6639-01-J

このマニュアルに記載されている仕様および製品に関する情報は、予告なしに変更されることがあります。このマニュアルに記載されている表現、情報、および推奨事項は、すべて正確であると考えていますが、明示的であれ黙示的であれ、一切の保証の責任を負わないものとします。このマニュアルに記載されている製品の使用は、すべてユーザ側の責任になります。

対象製品のソフトウェア ライセンスおよび限定保証は、製品に添付された『Information Packet』に記載されています。添付されていない場合には、代理店にご連絡ください。

FCC クラス A 適合装置に関する記述：この装置はテスト済みであり、FCC ルール Part 15 に規定された仕様のクラス A デジタル装置の制限に適合していることが確認済みです。これらの制限は、商業環境で装置を使用したときに、干渉を防止する適切な保護を規定しています。この装置は、無線周波エネルギーを生成、使用、または放射する可能性があり、この装置のマニュアルに記載された指示に従って設置および使用しなかった場合、ラジオおよびテレビの受信障害が起こることがあります。住宅地でこの装置を使用すると、干渉を引き起こす可能性があります。その場合には、ユーザ側の負担で干渉防止措置を講じる必要があります。

FCC クラス B 適合装置に関する記述：このマニュアルに記載された装置は、無線周波エネルギーを生成および放射する可能性があります。シスコシステムズの指示する設置手順に従わずに装置を設置した場合、ラジオおよびテレビの受信障害が起こることがあります。この装置はテスト済みであり、FCC ルール Part 15 に規定された仕様のクラス B デジタル装置の制限に適合していることが確認済みです。これらの仕様は、住宅地で使用したときに、このような干渉を防止する適切な保護を規定したものです。ただし、特定の設置条件において干渉が起きないことを保証するものではありません。

シスコシステムズの書面による許可なしに装置を改造すると、装置がクラス A またはクラス B のデジタル装置に対する FCC 要件に適合しなくなることがあります。その場合、装置を使用するユーザの権利が FCC 規制により制限されることがあり、ラジオまたはテレビの通信に対するいかなる干渉もユーザ側の負担で矯正するように求められることがあります。

装置の電源を切ることによって、この装置が干渉の原因であるかどうかを判断できます。干渉がなくなれば、シスコシステムズの装置またはその周辺機器が干渉の原因になっていると考えられます。装置がラジオまたはテレビ受信に干渉する場合には、次の方法で干渉が起きないようにしてください。

・干渉がなくなるまで、テレビまたはラジオのアンテナの向きを変えます。

・テレビまたはラジオの左右どちらかの側に装置を移動させます。

・テレビまたはラジオから離れたところに装置を移動させます。

・テレビまたはラジオとは別の回路にあるコンセントに装置を接続します（装置とテレビまたはラジオがそれぞれ別個のブレーカーまたはヒューズで制御されるようにします）。

米国シスコシステムズ社では、この製品の変更または改造を認めていません。変更または改造した場合には、FCC 認定が無効になり、さらに製品を操作する権限を失うことになります。

シスコシステムズが採用している TCP ヘッダー圧縮機能は、UNIX オペレーティング システムの UCB（University of California, Berkeley）パブリック ドメイン バージョンの一部として、UCB が開発したプログラムを最適化したものです。

ここに記載されている他のいかなる保証にもよらず、各社のすべてのマニュアルおよびソフトウェアは、障害も含めて「現状のまま」として提供されます。シスコシステムズおよびこれら各社は、商品性や特定の目的への適合性、権利を侵害しないことに関する、または取り扱い、使用、または取引によって発生する、明示されたまたは黙示された一切の保証の責任を負わないものとします。

いかなる場合においても、シスコシステムズおよびその代理店は、このマニュアルの使用またはこのマニュアルを使用できないことによって起こる制約、利益の損失、データの損傷など間接的で偶発的に起こる特殊な損害のあらゆる可能性がシスコシステムズまたは代理店に知らされていても、それらに対する責任を一切負いかねます。

CCSP、CCVP、Cisco Square Bridge のロゴ、Follow Me Browsing、StackWise は、Cisco Systems, Inc. の商標です。Changing the Way We Work, Live, Play, and Learn、iQuick Study は、Cisco Systems, Inc. のサービスマークです。Access Registrar、Aironet、ASIST、BPX、Catalyst、CCDA、CCDP、CCIE、CCIP、CCNA、CCNP、Cisco、Cisco Certified Internetwork Expert のロゴ、Cisco IOS、Cisco Press、Cisco Systems、Cisco Systems Capital、Cisco Systems のロゴ、Cisco Unity、Empowering the Internet Generation、Enterprise/Solver、EtherChannel、EtherFast、EtherSwitch、Fast Step、FormShare、GigaDrive、GigaStack、HomeLink、Internet Quotient、IOS、IP/TV、iQ Expertise、iQ のロゴ、iQ Net Readiness Scorecard、LightStream、Linksys、MeetingPlace、MGX、Networkers のロゴ、Networking Academy、Network Registrar、*Packet*、PIX、Post-Routing、Pre-Routing、ProConnect、RateMUX、ScriptShare、SlideCast、SMARTnet、StrataView Plus、TeleRouter、The Fastest Way to Increase Your Internet Quotient、VCO は、米国および一部の国における Cisco Systems, Inc. または関連会社の登録商標です。

このマニュアルまたは Web サイトで言及している他の商標はいずれも、それぞれの所有者のものです。「パートナー」という用語を使用していても、シスコシステムズと他社とのパートナー関係を意味するものではありません。（0502R）

Catalyst 6500 シリーズ スイッチ コンテント スイッチング モジュール コンフィギュレーション ノート

ソフトウェア ライセンス契約

本契約の各国語版は、シスコシステムズ（以下「シスコ」）のリセラーにお問い合わせいただくか、シスコの Web サイト www.cisco.com にアクセスしてください。シスコのソフトウェアまたはシスコが供給するソフトウェアをダウンロード、インストール、または使用される前に、本ソフトウェア ライセンス契約をよくお読みください。本ソフトウェアのダウンロード、インストール、または、本ソフトウェアを内蔵する機器の使用により、お客様は本契約の内容に同意したものとみなされます。本契約のいずれかの条項に同意されない場合には、本ソフトウェアのダウンロード、インストール、または使用を行わないでください。この場合お客様は、本ソフトウェアを返却および代金全額の払戻を受けるか、または本ソフトウェアが他の製品の一部として供給された場合には当該製品全体を返却して代金全額の払戻を受けることができます。返却および代金払戻は、シスコまたはシスコのリセラーから本ソフトウェアを購入後 30 日間有効であり、お客様が正規の購入者である場合のみ適用されます。

本ソフトウェアの使用には、(a) 特定のプログラムがシスコとの間で別途締結される書面による合意の対象となっている範囲、または (b) 特定のプログラムにおいてインストールの過程の一部として含まれる「クリック オン」ライセンス契約を除き、以下の条件が適用されます。

ライセンス　本契約に別途規定がある場合を除き、シスコシステムズ（以下「シスコ」）およびそのサプライヤは、本契約の条件に従って、お客様（以下「お客様」）に対し、お客様が規定のライセンス料を支払った特定のシスコのプログラム モジュール、フィーチャ セット、またはフィーチャ（以下「本ソフトウェア」）をオブジェクト コード形式でのみ使用する、非独占的かつ譲渡不能なライセンスを許諾します。また、上記ライセンスには、以下の各制限が適用されます。

- お客様は、書面に別途明示された場合を除き、本ソフトウェアを、お客様の所有または貸借するシスコ機器に内蔵された状態で、かかるシスコ機器上でのみ使用するか、または（関連文書によってシスコ以外の機器へのインストールが許可されている場合には）かかるシスコ機器との通信だけに使用するものとします。
- お客様は、単一のハードウェア シャーシ、単一の中央演算処理装置、またはシスコに支払った規定のライセンス料に相当する数量のシャーシまたは中央演算処理装置においてのみ、本ソフトウェアを使用できるものとします。
- また、お客様の本ソフトウェアの使用には、発行済み IP アドレス数、中央演算処理装置の性能、ポート数の制限に加え、本ソフトウェアに関するシスコの製品カタログに記載されているその他の制限が適用されます。

注：シスコがライセンス料を徴収しない評価版またはベータ版については、上記のライセンス料の支払要件は適用されません。

一般的な制限　お客様は、本契約に基づいて明示的に規定された場合を除き、以下の行為を行う権利はなく、また以下の行為を行わないことに同意します。(i) お客様のライセンスの他者への譲渡またはサブライセンス供与、および未承諾または中古品のシスコ機器での本ソフトウェアの使用。かかる譲渡またはサブライセンスの供与は無効とみなされます。(ii) 本ソフトウェアのエラー修正または他の方法での改変、本ソフトウェアに基づく派生物の作成、または第三者への当該行為の許可、(iii) 本ソフトウェアに含まれる企業秘密または極秘情報の取得を目的とした、本ソフトウェアのデコンパイル、復号化、リバース エンジニアリング、逆アセンブル、または他の方法による本ソフトウェアの判読可能形式への変換。シスコは、法律で要求される範囲において、お客様の要請に応じて、お客様が相応のシスコ料金を支払った場合、本ソフトウェアと独自に開発された他のプログラムとの互換性を得るために必要なインターフェイス情報を、お客様に提供するものとします。お客様は当該情報について、厳重な秘密保持義務を負うものとします。

アップグレードおよび追加コピー　本契約で言及する「ソフトウェア」には、シスコから、またはお客様が相応のライセンス料を支払った認定代理店からお客様にライセンス許諾または提供された本ソフトウェアのアップグレード版、アップデート版、バグ修正版または修正版（以下「アップグレード」）、およびバックアップ用の追加コピーが含まれるものとし、かかるアップグレードおよび追加コピーには、本契約の条件が適用されるものとします。本契約の他の規定に関係なく、(1) お客様が、かかる追加コピーまたはアップグレードの取得時に、正規のソフトウェアの有効ライセンスを保持し、アップグレードに必要な料金を支払っている場合を除き、お客様にはかかる追加コピーまたはアップグレードを使用するライセンスまたは権利はなく、(2) アップグレードの使用は、お客様が正規のエンドユーザ購入者または賃借者であるか、またはアップグレードされたソフトウェアに対して有効なライセンスを保持しているシスコ機器に限定され、(3) 追加コピーの使用は、バックアップ用途のみに限定されます。

所有権の表示　お客様は、いかなる形式であれ、本ソフトウェアのすべてのコピーについて、本ソフトウェアに含まれている著作権および他の登録商標権の表示を、本ソフトウェアと同じ形式かつ方法で保持し、複製することに同意します。本契約で明示的に認可されている場合を除き、お客様は、シスコから事前に書面による許可を得ることなく、本ソフトウェアのコピー、複製、またはソフトウェアそのものを作成しないものとします。お客様は、お客様の合法的な使用に必要な場合、本ソフトウェアのバックアップ コピーを作成することができます。ただし、かかるコピーには、オリジナル版に含まれるすべての著作権、機密保持、および登録商標権の表示を添付するものとします。

情報の保護　お客様は、個々のプログラムに特定の設計および構造を含め、本ソフトウェアおよび関連文書はすべて、シスコの企業秘密または著作権保護の対象であることに同意します。お客様は、シスコから事前に書面による承諾を得ることなく、かかる企業秘密または著作権保護された内容を、いかなる形式であっても、第三者に開示、提供、または利用させないものとします。お客様は、かかる企業秘密および著作権保護された内容に対して、妥当な保護策を講じるものとします。本ソフトウェアおよび関連文書の所有権は、シスコが単独で保有するものとします。

限定保証　お客様が本ソフトウェアをシスコから直接取得された場合、シスコは、(以下に定義する) 保証期間にわたり、(i) 一般的な使用下では、本ソフトウェアの媒体に材質上および製造上の欠陥がないこと、および (ii) 本ソフトウェアが公示仕様に実質的に適合していることを保証します。保証期間とは、お客様による本ソフトウェアの受領日から開始され、シスコからの本ソフトウェアの初回出荷日から 90 日後、または司法管轄区の法律が規定する最小期間最終日のどちらか遅い日付をもって終了する期間を意味します。また、本ソフトウェアには、シスコの西暦 2000 年限定保証が適用されることがあります。本ソフトウェアがこの保証の対象であるかどうかは、www.cisco.com/warp/public/779/smbiz/service/y2k/y2k_comp.htm を参照してください。限定保証が適用されるのは、正規にライセンスを供与されたお客様だけです。これらの限定保証に基づくお客様への唯一の保証、およびシスコとそのサプライヤの全責任は、シスコまたはサービス センターの判断により、本ソフトウェアの修理、交換、または返金（要求に応じて、シスコまたはその代理者に報告した場合）になります。本契約で明示的に供与されている場合を除き、本ソフトウェアは「現状のまま」提供されます。シスコは、本ソフトウェアにエラーが発生しないこと、またはお客様が本ソフトウェアを支障または障害なく使用できることを、保証しません。また、ネットワークへの侵入または攻撃の新技術は、日々開発されているため、シスコは、本ソフトウェアまたは本ソフトウェアを使用する機器、システムまたはネットワークがかかる侵入または攻撃を受けないことを保証しません。本保証は、(a) 本ソフトウェアのライセンスが、シスコがライセンス料を徴収しないベータ版、評価版、試験版、またはデモンストレーション用として供与されている場合、(b) 本ソフトウェアがシスコ以外で改変された場合、(c) シスコ提供のマニュアルに従ってインストール、運用、修理、または保守されなかった場合、(d) 本ソフトウェアが、過剰な物理的または電気的負荷、誤使用、不注意、または事故による障害を受けた場合、または (e) 本ソフトウェアが非常に危険な環境で使用された場合には、適用されません。お客様が本ソフトウェアをシスコのリセラーから購入された場合には、かかる販売業者が提供する保証条件が適用されるものとし、シスコはお客様に対して、本ソフトウェアに関する保証を提供しません。

保証の否認　本保証に明記されている場合を除き、商品性、特定用途への適合性、侵害の不在、および十分な品質に関する黙示保証 / 条件、または取引、使用、売買の過程で発生する黙示保証 / 条件を含みますがこれらに限定されず、一切の明示的または黙示の条件、表明、および保証は、適用法によって許可される範囲において、除外されるものとします。除外できない黙示保証については、かかる保証は本保証期間内に制限されます。州または司法管轄区によっては、黙示保証の有効期間を限定することが許可されていないため、お客様に上記の制限が適用されない場合があります。この保証はお客様に特別な法的権利を付与するもので、お客様は、司法管轄区によって異なるその他の権利を有する場合もあります。責任の否認。シスコまたはそのサプライヤは、本ソフトウェアの使用または使用不能によって発生した収益、利益、またはデータの損失、あるいは特殊、間接的、結果的、偶発的、または懲罰的な損害について、いかなる原因であれ、また責任理論に関係なく、シスコまたはそのサプライヤがかかる損害の可能性を通知されていたとしても、一切責任を負いません。契約上、不法行為（過失を含む）、その他に関するシスコまたはそのサプライヤのお客様に対する責任は、いかなる場合にも、お客様が支払った金額を超えないものとします。上記の制限は、上記の保証がその本来の目的を達成できない場合にも適用されます。州または司法管轄区によっては、結果的または偶発的な損害の制限または除外が許可されていないため、お客様に上記の制限が適用されない場合もあります。

期間および終了　本契約は、終了するまで有効です。お客様は、関連文書を含む本ソフトウェアのすべての複製物を廃棄することにより、本契約をいつでも終了させることができます。本契約に基づくお客様のライセンス権利は、お客様が本契約のいずれかの規程に従わない場合、シスコからの通告なしに、ただちに終了します。本契約の終了時に、お客様は、お客様が保有または管理する本ソフトウェアのすべての複製物を廃棄する必要があります。

お客様の記録　お客様はシスコとその独立会計士に対して、お客様の通常の営業時間中に、お客様の帳簿、記録、会計簿を査察し、本契約の条件に従っているかどうかを確認する権利を認めるものとします。

輸出　本ソフトウェアは、技術データを含め、米国輸出管理法を含む米国輸出規制法の対象となります。また、他国の輸出入規制の対象になることがあります。お客様は、かかる規制のすべてを厳密に遵守することに同意し、また、本ソフトウェアを輸出、再輸出、または輸入する場合にはライセンスを取得する責任があることを認知するものとします。

制限付権利　シスコの商用ソフトウェアおよび商用コンピュータ ソフトウェア文書は、本契約の条件および FAR（連邦調達規則）522.227-19（1987 年 6 月）第 (c) 項「商用コンピュータ ソフトウェア ― 制限付権利」に従って米国政府当局に提供されます。米国国防総省の機関への提供については、DFARS（国防連邦調達規則補則）252.227-7015（1995 年 11 月）の「技術データ ― 商用品目」に定める制限が適用されます。総則。本契約は、米国カリフォルニア州の州内で完全に執行されたものとみなされ、法の抵触の原則には一切影響せずに、当該州法に従って管理され、解釈されるものとします。本契約の一部が無効または施行不能になったとしても、本契約における他の条項の有効性は完全に保持されるものとします。シスコは本契約のもとに、国際物品売買契約に関する国連条約を明示的に否認します。本契約に明記されている場合を除き、本契約は、本ソフトウェアのライセンスに関する両当事者間の完全な合意を成すものとし、発注書に含まれる相反する条件または追加条件よりも優先されるものとします。

CONTENTS

FIGURES

はじめに

ここでは、『*Catalyst 6500 シリーズ スイッチ コンテント スイッチング モジュール コンフィギュレーション ノート*』の対象読者、マニュアルの構成、および手順や情報を記述するための表記法について説明します。

（注） 特に区別されていないかぎり、「Catalyst 6500 シリーズ スイッチ」には、Catalyst 6500 シリーズと Catalyst 6000 シリーズの両スイッチが含まれます。

このマニュアルでは、Catalyst 6500 シリーズ スイッチ シャーシの設置手順については説明しません。スイッチ シャーシの設置手順については、『*Catalyst 6500 Series Switch Installation Guide*』を参照してください。

（注） 警告については、「安全に関する概要」（p.xviii）を参照してください。

対象読者

このマニュアルに記載された装置の設置、交換、またはサービスは、訓練を受けた認定サービス技術者（IEC 60950 および AS/NZS3260 で定義）だけが行ってください。

マニュアルの構成

このマニュアルの構成は、次のとおりです。

章	タイトル	説明
第 1 章	製品概要	Catalyst 6500 シリーズ Content Switching Module（CSM; コンテント スイッチング モジュール）について概要を説明します。
第 2 章	CSM によるネットワーキング	ネットワーク上での CSM の動作について説明します。
第 3 章	設定前の作業	CSM 上でコンテント スイッチングを実行するためのクイック スタート ガイドを示します。
第 4 章	VLAN の設定	CSM にクライアントおよびサーバ VLAN（仮想 LAN）を設定する方法について説明します。
第 5 章	実サーバおよびサーバ ファームの設定	CSM 上でロードバランシングを設定する方法について説明します。
第 6 章	仮想サーバ、マップ、およびポリシーの設定	CSM 上でヘルス モニタリングを設定する方法について説明します。
第 7 章	冗長性の設定	フォールト トレランス、HSRP、接続の冗長性、およびヒットレス アップグレードの設定方法について説明します。
第 8 章	追加機能およびオプションの設定	固定グループおよび Route Health Injection（RHI）、Global Server Load Balancing（GSLB; グローバル サーバ ロードバランシング）、およびネットワーク管理の設定方法について説明します。
第 9 章	ヘルス モニタリングの設定	サーバおよびサーバ ファームのヘルスを設定およびモニタする方法について説明します。
第 10 章	CSM での TCL スクリプトの使用	Toolkit Command Language（TCL）スクリプトを使用して CSM を設定する方法について説明します。
第 11 章	ファイアウォール ロードバランシングの設定	CSM によるロードバランシング設定でのファイアウォールについて説明します。
付録 A	コンフィギュレーション例	CSM のコンフィギュレーションのサンプルを示します。
付録 B	トラブルシューティングとシステム メッセージ	トラブルシューティング情報を提供し、システム メッセージを示します。
付録 C	CSM XML の DTD	CSM のエラー メッセージとともに、エラーの原因についての説明および問題解決に必要な操作について示します。

表記法

このマニュアルでは、次の表記法を使用しています。

表記	説明
太字	コマンド、コマンド オプション、およびキーワードは**太字**で示しています。
イタリック体	ユーザが値を指定する引数は、*イタリック体* で示しています。
[ ]	角カッコの中の要素は、省略可能です。
{ x | y | z }	必ずどれか 1 つを選択しなければならない必須キーワードは、波カッコで囲み、縦棒で区切って示しています。
[x | y | z]	どれか 1 つを選択できる省略可能なキーワードは、角カッコで囲み、縦棒で区切って示しています。
ストリング	引用符を付けない一組の文字。ストリングの前後には引用符を使用しません。引用符を使用すると、その引用符も含めてストリングとみなされます。
`screen` フォント	システムが表示する端末セッションおよび情報は、`screen` フォントで示しています。
太字の **`screen`** フォント	ユーザが入力しなければならない情報は、**太字**の **`screen`** フォントで示しています。
*イタリック体*の *`screen`* フォント	ユーザが値を指定する引数は、*イタリック体* の *`screen`* フォントで示しています。
^	^ 記号は、Ctrl キーを表します。たとえば、画面に表示される ^D というキーの組み合わせは、Ctrl キーを押しながら D キーを押すことを意味します。
< >	パスワードのように出力されない文字は、かぎカッコ（<>）で囲んで示しています。

（注）は、次のように表しています。

（注） 「*注釈*」です。役立つ情報や、このマニュアル以外の参照資料などを紹介しています。

ヒントは、次のように表しています。

ヒント 「*問題解決に役立つ情報*」です。ヒントには、トラブルシューティングや操作方法ではなく、ワンポイントアドバイスと同様に知っておくと役立つ情報が記述される場合もあります。

注意は、次のように表しています。

注意 「*要注意*」の意味です。機器の損傷またはデータ損失を予防するための注意事項が記述されています。

安全に関する概要

誤って行うと危険が生じる可能性のある操作については、安全上の警告が記載されています。各警告文に、警告を表す記号が記されています。

警告　**安全上の重要事項**

「危険」の意味です。人身事故を予防するための注意事項が記述されています。機器の取り扱い作業を行うときは、電気回路の危険性に注意し、一般的な事故防止対策に留意してください。

これらの注意事項を保存しておいてください。

関連資料

Content Switching Module に対するインストレーションおよびコンフィギュレーションの詳細は、次の関連資料を参照してください。

- 『*Release Notes for the Catalyst 6500 Series Switch Content Switching Module*』
- 『*Catalyst 6500 Series Switch Content Switching Module Installation Note*』
- 『*Catalyst 6500 Series Switch Content Switching Module Command Reference*』
- 『*Regulatory Compliance and Safety Information for the Catalyst 6500 Series Switches*』

インストレーションおよびコンフィギュレーションの詳細は、次の関連資料を参照してください。

- 『*Catalyst 6500 Series Switch Installation Guide*』
- 『*Catalyst 6500 Series Switch Quick Software Configuration Guide*』
- 『*Catalyst 6500 Series Switch Module Installation Guide*』
- 『*Catalyst 6500 Series Switch Software Configuration Guide*』
- 『*Catalyst 6500 Series Switch Command Reference*』
- 『*Catalyst 6500 Series Switch Cisco IOS Software Configuration Guide*』
- 『*Catalyst 6500 Series Switch Cisco IOS Command Reference*』
- 『*ATM Software Configuration and Command Reference ― Catalyst 5000 Family and Catalyst 6500 Series Switches*』
- 『*System Message Guide ― Catalyst 6500 Series Switches*』
- MIB（管理情報ベース）については、次の URL を参照してください。

  http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml
- *Catalyst 6500 シリーズ スイッチおよびCisco IOS Release 12.1(8a)E3 対応Cisco 7600 シリーズ ルータのリリース ノート*

Cisco IOS のコンフィギュレーション ガイドおよびコマンド リファレンス ― MSFC 上および MSM/ATM モジュール上で稼働する Cisco IOS ソフトウェアを設定するときに役立ちます。

マニュアルの入手方法

シスコの製品マニュアルおよびその他の資料は、Cisco.com で入手できます。また、テクニカル サポートおよびその他のリソースもさまざまな方法で入手できます。ここでは、シスコ製品に関する技術情報を入手する方法について説明します。

Cisco.com

次の URL から、シスコ製品の最新資料を入手することができます。

http://www.cisco.com/univercd/home/home.htm

シスコの Web サイトには、次の URL からアクセスしてください。

http://www.cisco.com

シスコの Web サイトの各国語版へは、次の URL からアクセスできます。

http://www.cisco.com/public/countries_languages.shtml

Documentation DVD

シスコ製品のマニュアルおよびその他の資料は、製品に付属の Documentation DVD パッケージでご利用いただけます。Documentation DVD は定期的に更新されるので、印刷資料よりも新しい情報が得られます。この DVD パッケージは、単独で入手することができます。

Cisco.com（Cisco Direct Customers）に登録されている場合、Ordering ツールまたは Cisco Marketplace から Cisco Documentation DVD（Customer Order Number DOC-DOCDVD=）を発注できます。

Cisco Ordering ツール：

http://www.cisco.com/en/US/partner/ordering/

Cisco Marketplace：

http://www.cisco.com/go/marketplace/

マニュアルの発注方法

マニュアルの発注方法については、次の URL にアクセスしてください。

http://www.cisco.com/univercd/cc/td/doc/es_inpck/pdi.htm

シスコ製品のマニュアルは、次の方法でご発注いただけます。

- Cisco.com（Cisco Direct Customers）に登録されている場合、Ordering ツールからシスコ製品のマニュアルを発注できます。次の URL にアクセスしてください。

  http://www.cisco.com/en/US/partner/ordering/
- Cisco.com に登録されていない場合、製品を購入された代理店へお問い合わせください。

シスコ製品のセキュリティ

シスコでは、無償の Security Vulnerability Policy ポータルを次の URL で提供しています。

http://www.cisco.com/en/US/products/products_security_vulnerability_policy.html

このサイトから、以下のタスクを実行できます。

- シスコ製品における脆弱性を報告する。
- シスコ製品のセキュリティ問題に対する支援を受ける。
- シスコからのセキュリティ情報を入手するために登録を行う。

シスコ製品に関するセキュリティ勧告および注意のリストが以下の URL で確認できます。

http://www.cisco.com/go/psirt

勧告および注意事項が変更された際に、リアルタイムで確認したい場合は、以下の URL から Product Security Incident Response Team Really Simple Syndication（PSIRT RSS）にアクセスできます。

http://www.cisco.com/en/US/products/products_psirt_rss_feed.html

シスコ製品のセキュリティ問題の報告

シスコでは、安全な製品を提供することを目指しています。製品のリリース前に社内でテストを実施し、すべての脆弱性を迅速に修正するように努めております。お客様がシスコ製品の脆弱性を発見したと思われる場合は、次の PSIRT にご連絡ください。

- 緊急度の高い問題 — security-alert@cisco.com
- 緊急度の低い問題 — psirt@cisco.com

ヒント お客様が第三者に知られたくない情報をシスコに送信する場合、Pretty Good Privacy（PGP）または PGP と互換性のある製品を使用して情報を暗号化することを推奨します。PSIRT は、PGP バージョン 2.*x* ～ 8.*x* と互換性のある暗号化情報を取り扱うことができます。

無効な暗号鍵または失効した暗号鍵は使用しないでください。PSIRT と通信する際は、次の公開鍵サーバの一覧に記載されている有効な公開鍵を使用してください。

http://pgp.mit.edu:11371/pks/lookup?search=psirt%40cisco.com&op=index&exact=on

緊急度の高い問題の場合、次の電話番号で PSIRT に問い合わせることができます。

- 1 877 228-7302
- 1 408 525-6532

テクニカル サポート

Cisco Technical Support では、シスコシステムズとサービス契約を結んでいるお客様、パートナー、リセラー、販売店を対象として、評価の高い 24 時間体制のテクニカル サポートを提供しています。Cisco.com の Cisco Technical Support Web サイトでは、広範囲にわたるオンラインでのサポート リソースを提供しています。さらに、Technical Assistance Center（TAC）では、電話でのサポートも提供しています。シスコシステムズとサービス契約を結んでいない場合は、リセラーにお問い合わせください。

Cisco Technical Support Web サイト

Cisco Technical Support Web サイトでは、オンラインで資料やツールを利用して、トラブルシューティングやシスコ製品およびテクノロジーに関する技術上の問題の解決に役立てることができます。Cisco Technical Support Web サイトは、1 年中いつでも利用することができます。次の URL にアクセスしてください。

http://www.cisco.com/techsupport

Cisco Technical Support Web サイト上のツールにアクセスする際は、いずれも Cisco.com のログイン ID およびパスワードが必要です。サービス契約が有効で、ログイン ID またはパスワードを取得していない場合は、次の URL で登録手続きを行ってください。

http://tools.cisco.com/RPF/register/register.do

（注） テクニカル サポートにお問い合わせいただく前に、Cisco Product Identification（CPI）ツールを使用して、製品のシリアル番号をご確認ください。CPI ツールへは、Documentation & Tools の下にある Tools & Resources リンクをクリックして、Cisco Technical Support Web サイトからアクセスできます。Alphabetical Index ドロップダウン リストから **Cisco Product Identification Tool** を選択するか、Alerts & RMAs の下にある **Cisco Product Identification Tool** リンクをクリックしてください。CPI ツールは、製品 ID またはモデル名、ツリー表示、または特定の製品に対する **show** コマンド出力のコピー＆ペーストによる 3 つの検索オプションを提供します。検索結果には、シリアル番号のラベルの場所がハイライトされた製品の説明図が表示されます。テクニカル サポートにお問い合わせいただく前に、製品のシリアル番号のラベルを確認し、メモなどに控えておいてください。

Japan TAC Web サイト

Japan TAC Web サイトでは、利用頻度の高い TAC Web サイト（http://www.cisco.com/tac）のドキュメントを日本語で提供しています。Japan TAC Web サイトには、次の URL からアクセスしてください。

http://www.cisco.com/jp/go/tac

サポート契約を結んでいない方は、「ゲスト」としてご登録いただくだけで、Japan TAC Web サイトのドキュメントにアクセスできます。

Japan TAC Web サイトにアクセスするには、Cisco.com のログイン ID とパスワードが必要です。ログイン ID とパスワードを取得していない場合は、次の URL にアクセスして登録手続きを行ってください。

http://www.cisco.com/jp/register/

Service Request ツールの使用

オンラインの TAC Service Request ツールを使えば、S3 および S4 の問題について最も迅速にテクニカル サポートを受けられます（ネットワークの障害が軽微である場合、あるいは製品情報が必要な場合）。状況をご説明いただくと、TAC Service Request ツールが推奨される解決方法を提供します。これらの推奨リソースを使用しても問題が解決しない場合は、TAC の技術者が対応します。TAC Service Request ツールは次の URL からアクセスできます。

http://www.cisco.com/techsupport/servicerequest

問題が S1 または S2 であるか、インターネットにアクセスできない場合は、電話で TAC にご連絡ください（運用中のネットワークがダウンした場合、あるいは重大な障害が発生した場合）。S1 および S2 の問題には TAC の技術者がただちに対応し、業務を円滑に運営できるよう支援します。

電話でテクニカル サポートを受ける際は、次の番号のいずれかをご使用ください。

アジア太平洋：+61 2 8446 7411（オーストラリア：1 800 805 227）
EMEA：+32 2 704 55 55
米国：1 800 553-2447

TAC の連絡先一覧については、次の URL にアクセスしてください。

http://www.cisco.com/techsupport/contacts

問題の重大度の定義

すべての問題を標準形式で報告するために、問題の重大度を定義しました。

重大度 1（S1）— ネットワークがダウンし、業務に致命的な損害が発生する場合。24 時間体制であらゆる手段を使用して問題の解決にあたります。

重大度 2（S2）— ネットワークのパフォーマンスが著しく低下、またはシスコ製品のパフォーマンス低下により業務に重大な影響がある場合。通常の業務時間内にフルタイムで問題の解決にあたります。

重大度 3（S3）— ネットワークのパフォーマンスが低下しているが、ほとんどの業務運用が機能している場合。通常の業務時間内にサービスの復旧を行います。

重大度 4（S4）— シスコ製品の機能、インストレーション、基本的なコンフィギュレーションについて、情報または支援が必要で、業務への影響がほとんどまたはまったくない場合。

その他の資料および情報の入手方法

シスコの製品、テクノロジー、およびネットワーク ソリューションに関する情報について、さまざまな資料をオンラインおよび印刷物で入手することができます。

- Cisco Marketplace は、さまざまなシスコの書籍、参考資料、およびロゴ入り商品を提供しています。Cisco Marketplace には、次の URL からアクセスしてください。

  http://www.cisco.com/go/marketplace/

- Cisco Press では、ネットワーク、トレーニング、認定関連の出版物を幅広く発行しています。初心者から上級者まで、さまざまな読者向けの出版物があります。Cisco Press の最新の出版情報などについては、次の URL からアクセスしてください。

  http://www.ciscopress.com

- 『*Packet*』は、シスコシステムズが発行するテクニカル　ユーザ向けの季刊誌で、インターネットやネットワークへの投資を最大限に活用するのに役立ちます。『*Packet*』には、ネットワーク分野の最新動向、テクノロジーの進展、およびシスコの製品やソリューションに関する記事をはじめ、ネットワークの配置やトラブルシューティングのヒント、設定例、お客様の事例研究、認定やトレーニングに関する情報、および多数の詳細なオンライン リソースへのリンクが盛り込まれています。『*Packet*』には、次の URL からアクセスしてください。

  http://www.cisco.com/packet

- 『*iQ Magazine*』は、シスコのテクノロジーを使って収益の増加、ビジネス効率の向上、およびサービスの拡大を図る方法について学ぶことを目的とした、シスコシステムズが発行する成長企業向けの季刊誌です。この季刊誌は、実際の事例研究や事業戦略を用いて、これら企業が直面するさまざまな課題や、問題解決の糸口となるテクノロジーを明確化し、テクノロジーの投資に関して読者が正しい決断を行う手助けをします。『*iQ Magazine*』には、次の URL からアクセスしてください。

  http://www.cisco.com/go/iqmagazine

- 『*Internet Protocol Journal*』は、インターネットおよびイントラネットの設計、開発、運用を担当するエンジニア向けに、シスコシステムズが発行する季刊誌です。『*Internet Protocol Journal*』には、次の URL からアクセスしてください。

  http://www.cisco.com/ipj

- シスコシステムズは最高水準のネットワーク関連のトレーニングを実施しています。トレーニングの最新情報については、次の URL からアクセスしてください。

  http://www.cisco.com/en/US/learning/index.html

ライセンス

以下は、ソフトウェア ライセンスに関する情報です。

ソフトウェア ライセンス契約

本契約の各国語版は、シスコシステムズ（以下「シスコ」）のリセラーにお問い合わせいただくか、シスコの Web サイト www.cisco.com にアクセスしてください。シスコのソフトウェアまたはシスコが供給するソフトウェアをダウンロード、インストール、または使用される前に、本ソフトウェア ライセンス契約をよくお読みください。本ソフトウェアのダウンロード、インストール、または、本ソフトウェアを内蔵する機器の使用により、お客様は本契約の内容に同意したものとみなされます。本契約のいずれかの条項に同意されない場合には、本ソフトウェアのダウンロード、インストール、または使用を行わないでください。この場合お客様は、本ソフトウェアを返却および代金全額の払戻を受けるか、または本ソフトウェアが他の製品の一部として供給された場合には当該製品全体を返却して代金全額の払戻を受けることができます。返却および代金払戻は、シスコまたはシスコのリセラーから本ソフトウェアを購入後 30 日間有効であり、お客様が正規の購入者である場合のみ適用されます。

本ソフトウェアの使用には、(a) 特定のプログラムがシスコとの間で別途締結される書面による合意の対象となっている範囲、または (b) 特定のプログラムにおいてインストールの過程の一部として含まれる「クリック オン」ライセンス契約を除き、以下の条件が適用されます。

ライセンス　本契約に別途規定がある場合を除き、シスコシステムズ（以下「シスコ」）およびそのサプライヤは、本契約の条件に従って、お客様（以下「お客様」）に対し、お客様が規定のライセンス料を支払った特定のシスコのプログラム モジュール、フィーチャ セット、またはフィーチャ（以下「本ソフトウェア」）をオブジェクト コード形式でのみ使用する、非独占的かつ譲渡不能なライセンスを許諾します。また、上記ライセンスには、以下の各制限が適用されます。

- お客様は、書面に別途明示された場合を除き、本ソフトウェアを、お客様の所有または貸借するシスコ機器に内蔵された状態で、かかるシスコ機器上でのみ使用するか、または（関連文書によってシスコ以外の機器へのインストールが許可されている場合には）かかるシスコ機器との通信だけに使用するものとします。
- お客様は、単一のハードウェア シャーシ、単一の中央演算処理装置、またはシスコに支払った規定のライセンス料に相当する数量のシャーシまたは中央演算処理装置においてのみ、本ソフトウェアを使用できるものとします。
- また、お客様の本ソフトウェアの使用には、発行済み IP アドレス数、中央演算処理装置の性能、ポート数の制限に加え、本ソフトウェアに関するシスコの製品カタログに記載されているその他の制限が適用されます。

注：シスコがライセンス料を徴収しない評価版またはベータ版については、上記のライセンス料の支払要件は適用されません。

一般的な制限　お客様は、本契約に基づいて明示的に規定された場合を除き、以下の行為を行う権利はなく、また以下の行為を行わないことに同意します。(i) お客様のライセンスの他者への譲渡またはサブライセンス供与、および未承諾または中古品のシスコ機器での本ソフトウェアの使用。かかる譲渡またはサブライセンスの供与は無効とみなされます。(ii) 本ソフトウェアのエラー修正または他の方法での改変、本ソフトウェアに基づく派生物の作成、または第三者への当該行為の許可、(iii) 本ソフトウェアに含まれる企業秘密または極秘情報の取得を目的とした、本ソフトウェアのデコンパイル、復号化、リバース エンジニアリング、逆アセンブル、または他の方法による本ソフトウェアの判読可能形式への変換。シスコは、法律で要求される範囲において、お客様の要請に応じて、お客様が相応のシスコ料金を支払った場合、本ソフトウェアと独自に開発された他のプログラムとの互換性を得るために必要なインターフェイス情報を、お客様に提供するものとします。お客様は当該情報について、厳重な秘密保持義務を負うものとします。

アップグレードおよび追加コピー　本契約で言及する「ソフトウェア」には、シスコから、またはお客様が相応のライセンス料を支払った認定代理店からお客様にライセンス許諾または提供された本ソフトウェアのアップグレード版、アップデート版、バグ修正版または修正版（以下「アップグレード」）、およびバックアップ用の追加コピーが含まれるものとし、かかるアップグレードおよび追加コピーには、本契約の条件が適用されるものとします。本契約の他の規定に関係なく、(1) お客様が、かかる追加コピーまたはアップグレードの取得時に、正規のソフトウェアの有効ライセンスを保持し、アップグレードに必要な料金を支払っている場合を除き、お客様にはかかる追加コピーまたはアップグレードを使用するライセンスまたは権利はなく、(2) アップグレードの使用は、お客様が正規のエンドユーザ購入者または賃借者であるか、またはアップグレードされたソフトウェアに対して有効なライセンスを保持しているシスコ機器に限定され、(3) 追加コピーの使用は、バックアップ用途のみに限定されます。

所有権の表示　お客様は、いかなる形式であれ、本ソフトウェアのすべてのコピーについて、本ソフトウェアに含まれている著作権および他の登録商標権の表示を、本ソフトウェアと同じ形式かつ方法で保持し、複製することに同意します。本契約で明示的に認可されている場合を除き、お客様は、シスコから事前に書面による許可を得ることなく、本ソフトウェアのコピー、複製、またはソフトウェアそのものを作成しないものとします。お客様は、お客様の合法的な使用に必要な場合、本ソフトウェアのバックアップ コピーを作成することができます。ただし、かかるコピーには、オリジナル版に含まれるすべての著作権、機密保持、および登録商標権の表示を添付するものとします。

情報の保護　お客様は、個々のプログラムに特定の設計および構造を含め、本ソフトウェアおよび関連文書はすべて、シスコの企業秘密または著作権保護の対象であることに同意します。お客様は、シスコから事前に書面による承諾を得ることなく、かかる企業秘密または著作権保護された内容を、いかなる形式であっても、第三者に開示、提供、または利用させないものとします。お客様は、かかる企業秘密および著作権保護された内容に対して、妥当な保護策を講じるものとします。本ソフトウェアおよび関連文書の所有権は、シスコが単独で保有するものとします。

限定保証　お客様が本ソフトウェアをシスコから直接取得された場合、シスコは、(以下に定義する）保証期間にわたり、(i) 一般的な使用下では、本ソフトウェアの媒体に材質上および製造上の欠陥がないこと、および (ii) 本ソフトウェアが公示仕様に実質的に適合していることを保証します。保証期間とは、お客様による本ソフトウェアの受領日から開始され、シスコからの本ソフトウェアの初回出荷日から 90 日後、または司法管轄区の法律が規定する最小期間最終日のどちらか遅い日付をもって終了する期間を意味します。また、本ソフトウェアには、シスコの西暦 2000 年限定保証が適用されることがあります。本ソフトウェアがこの保証の対象であるかどうかは、www.cisco.com/warp/public/779/smbiz/service/y2k/y2k_comp.htm を参照してください。限定保証が適用されるのは、正規にライセンスを供与されたお客様だけです。これらの限定保証に基づくお客様への唯一の保証、およびシスコとそのサプライヤの全責任は、シスコまたはサービス センターの判断により、本ソフトウェアの修理、交換、または返金（要求に応じて、シスコまたはその代理者に報告した場合）になります。本契約で明示的に供与されている場合を除き、本ソフトウェアは「現状のまま」提供されます。シスコは、本ソフトウェアにエラーが発生しないこと、またはお客様が本ソフトウェアを支障または障害なく使用できることを、保証しません。また、ネットワークへの侵入または攻撃の新技術は、日々開発されているため、シスコは、本ソフトウェアまたは本ソフトウェアを使用する機器、システムまたはネットワークがかかる侵入または攻撃を受けないことを保証しません。本保証は、(a) 本ソフトウェアのライセンスが、シスコがライセンス料を徴収しないベータ版、評価版、試験版、またはデモンストレーション用として供与されている場合、(b) 本ソフトウェアがシスコ以外で改変された場合、(c) シスコ提供のマニュアルに従ってインストール、運用、修理、または保守されなかった場合、(d) 本ソフトウェアが、過剰な物理的または電気的負荷、誤使用、不注意、または事故による障害を受けた場合、または (e) 本ソフトウェアが非常に危険な環境で使用された場合には、適用されません。お客様が本ソフトウェアをシスコのリセラーから購入された場合には、かかる販売業者が提供する保証条件が適用されるものとし、シスコはお客様に対して、本ソフトウェアに関する保証を提供しません。

保証の否認　本保証に明記されている場合を除き、商品性、特定用途への適合性、侵害の不在、および十分な品質に関する黙示保証 / 条件、または取引、使用、売買の過程で発生する黙示保証 / 条件を含みますがこれらに限定されず、一切の明示的または黙示の条件、表明、および保証は、適用法によって許可される範囲において、除外されるものとします。除外できない黙示保証については、かかる保証は本保証期間内に制限されます。州または司法管轄区によっては、黙示保証の有効期間を限定することが許可されていないため、お客様に上記の制限が適用されない場合があります。この保証はお客様に特別な法的権利を付与するもので、お客様は、司法管轄区によって異なるその他の権利を有する場合もあります。責任の否認。シスコまたはそのサプライヤは、本ソフトウェアの使用または使用不能によって発生した収益、利益、またはデータの損失、あるいは特殊、間接的、結果的、偶発的、または懲罰的な損害について、いかなる原因であれ、また責任理論に関係なく、シスコまたはそのサプライヤがかかる損害の可能性を通知されていたとしても、一切責任を負いません。契約上、不法行為（過失を含む）、その他に関するシスコまたはそのサプライヤのお客様に対する責任は、いかなる場合にも、お客様が支払った金額を超えないものとします。上記の制限は、上記の保証がその本来の目的を達成できない場合にも適用されます。州または司法管轄区によっては、結果的または偶発的な損害の制限または除外が許可されていないため、お客様に上記の制限が適用されない場合もあります。

期間および終了　本契約は、終了するまで有効です。お客様は、関連文書を含む本ソフトウェアのすべての複製物を廃棄することにより、本契約をいつでも終了させることができます。本契約に基づくお客様のライセンス権利は、お客様が本契約のいずれかの規程に従わない場合、シスコからの通告なしに、ただちに終了します。本契約の終了時に、お客様は、お客様が保有または管理する本ソフトウェアのすべての複製物を廃棄する必要があります。

お客様の記録　お客様はシスコとその独立会計士に対して、お客様の通常の営業時間中に、お客様の帳簿、記録、会計簿を査察し、本契約の条件に従っているかどうかを確認する権利を認めるものとします。

輸出　本ソフトウェアは、技術データを含め、米国輸出管理法を含む米国輸出規制法の対象となります。また、他国の輸出入規制の対象になることがあります。お客様は、かかる規制のすべてを厳密に遵守することに同意し、また、本ソフトウェアを輸出、再輸出、または輸入する場合にはライセンスを取得する責任があることを認知するものとします。

制限付権利　シスコの商用ソフトウェアおよび商用コンピュータ ソフトウェア文書は、本契約の条件および FAR（連邦調達規則）522.227-19（1987 年 6 月）第（c）項「商用コンピュータ ソフトウェア ― 制限付権利」に従って米国政府当局に提供されます。米国国防総省の機関への提供については、DFARS（国防連邦調達規則補則）252.227-7015（1995 年 11 月）の「技術データ ― 商用品目」に定める制限が適用されます。総則。本契約は、米国カリフォルニア州の州内で完全に執行されたものとみなされ、法の抵触の原則には一切影響せずに、当該州法に従って管理され、解釈されるものとします。本契約の一部が無効または施行不能になったとしても、本契約における他の条項の有効性は完全に保持されるものとします。シスコは本契約のもとに、国際物品売買契約に関する国連条約を明示的に否認します。本契約に明記されている場合を除き、本契約は、本ソフトウェアのライセンスに関する両当事者間の完全な合意を成すものとし、発注書に含まれる相反する条件または追加条件よりも優先されるものとします。

製品概要

Catalyst 6500 シリーズ Content Switching Module（CSM; コンテント スイッチング モジュール）は、レイヤ 3 ～ 7 までのパケット情報に基づき、サーバ グループ、サーバ ファーム、ファイアウォール、キャッシュ、VPN 終端デバイス、およびその他のネットワーク デバイスの間で負荷を分散する高性能な Server Load Balancing（SLB; サーバ ロードバランシング）を提供します。

サーバ ファームは、ロードバランスの対象装置からなるグループです。サーバ ファームを仮想サーバにすることによって、ネットワークのスケーラビリティとサービス アベイラビリティが向上します。仮想サーバのアベイラビリティに影響を与えることなく、いつでも新しいサーバを追加したり、故障したサーバまたは既存のサーバを除去したりできます。

クライアントを CSM に接続するには、仮想サーバの VIP（仮想 IP）アドレスに要求を送ります。クライアントが仮想サーバへの接続を開始すると、CSM は設定されたロードバランシング アルゴリズムおよびポリシー（アクセス ルール）に基づいて、接続用の実サーバ（サーバ ファームに割り当てられる物理装置）を選択します。ポリシーでは、クライアント接続の送り先を定義することによってトラフィックを管理します。

固定（sticky）接続は、送信元 IP アドレス、送信元 IP サブネット、Cookie、および Secure Socket Layer（SSL）を使用して、同じクライアントからの複数の接続を同じ実サーバに*固定*することによって、または HTTP リダイレクト メッセージを使用してこれらの接続をリダイレクトすることによって、個々のサーバへのトラフィックを制限します。

ここでは、CSM について説明します。

機能

今回のソフトウェア リリースには、旧リリースからの CSM 機能をサポートするフィーチャ セットが含まれます。ここでは、表形式で次のフィーチャ セットを示します。

表 1-1 に、このリリースの新しい CSM 機能を示します。

表 1-1　新しい CSM フィーチャ セットの説明

このリリースの新機能	説明
HTTP ヘッダーの固定	HTTP ヘッダーの内容（たとえば、MSISDN[1] 番号、サービス キー、セッション ID など）に基づいて固定処理を行うように CSM を設定できます。
設定の同期化	フォールトトレランス VLAN を経由したアクティブ CSM とスタンバイ CSM 間の設定の同期化をサポートします。
インターフェイスと重要なデバイスのフェールオーバー トラッキング	HSRP グループ、物理インターフェイス、およびゲートウェイの状態を追跡できます。
プライベート VLAN	CSM で Private VLAN（PVLAN; プライベート VLAN）を使用可能にします。
部分的なサーバ ファーム フェールオーバー	プライマリ サーバ ファームで部分的に障害が発生した場合に CSM がバックアップ サーバ ファームにフェールオーバーするように、バックアップ サーバ ファームの設定時にスレッシュホールドの値を定義できます。
サーバ プローブ失敗ステートの改善	プローブに失敗したサーバを復旧するために必要となる再試行回数を指定できます。
実名オプション	エンティティに関する詳細を指定できます。このオプションは、プローブ、vserver、VLAN、サーバファームの各モードに適用できます。
Network Address Translation（NAT; ネットワーク アドレス変換）設定の拡張機能	送信元 NAT（NAT クライアント）の設定ルールをポリシー レベルにまで向上させることができます。
無限アイドル タイムアウト	無期限で接続をオープンなまま維持できます。
VIP の依存関係	複数の VIP を一括してリンクし、指定した VIP が停止するとそれに従属する VIP も自動的に停止する機能を提供します。
ポリシーの順序	特定のポリシーにプライオリティ値を割り当てることができます。
解析可能な最大長に達したときの動作の変更	CSM は、解析可能な最大長の接続要求を複数のデフォルト ポリシーにロードバランシングします。
スロー スタートの改善	スロースタート タイマーが期限切れになるか、conn_count 値がその他の実サーバの conn_count 値と等しくなるまで、実サーバをスロースタート モードにしておくことができます。
非セキュア ルータ モード	SYN パケット以外に、VIP にヒットしない非 SYN パケットもルーティングするように環境変数が拡張されました。

表 1-1　新しい CSM フィーチャ セットの説明（続き）

このリリースの新機能	説明
vserver の制限数を増加	特定の VIP ごとに設定可能な仮想サーバの数が 128 から 1000 に増えています。
リモート デスクトップ プロトコル	MSTS-RDP を設定するための環境変数が追加されています。[2]

1. MSISDN = Mobile Station ISDN; モバイル ステーション ISDN
2. MSTS-RDP = Microsoft Terminal Services Remote Desktop Protocol

表 1-2 に、このリリースと旧リリースに対応している CSM 機能を示します。

表 1-2　CSM フィーチャ セットの説明

機能
サポート対象ハードウェア
Supervisor 1A（Multilayer Switch Feature Card［MSFC; マルチレイヤ スイッチ フィーチャ カード］および Policy Feature Card［PFC; ポリシー フィーチャ カード］内蔵）
Supervisor 2（MSFC および PFC 内蔵）
Supervisor 720 ― CSM ソフトウェア Release 3.1(4) 以上が必要です。
サポート対象プロトコル
TCP ロードバランシング
UDP 一般 IP プロトコル ロードバランシング
FTP および Real Time Streaming Protocol（RTSP）に関する特殊なアプリケーション レイヤ サポート
Server Application State Protocol（SASP）
レイヤ 7 機能
完全正規表現照合
URL、Cookie スイッチング、一般 HTTP ヘッダー解析、HTTP メソッド解析
その他の機能
VIP 接続のウォーターマーク
バックアップ（ソーリー サーバ）およびサーバ ファーム
ヘルス プローブ用のオプション ポート
IP 再組み立て
TCL スクリプト
XML コンフィギュレーション インターフェイス
SNMP（簡易ネットワーク管理プロトコル）
Global Server Load Balancing（GSLB; グローバル サーバ ロードバランシング）― ライセンスが必要です。
リソース使用状況の表示
設定可能なアイドルおよび保留接続タイムアウト
単一方向フローのアイドル タイムアウト
SSL ロードバランシングの STE 統合
実サーバ名
すべてのタイプのフロー（TCP、UDP、および IP）に関する TCP 接続の冗長性
フォールト トレラント **show** コマンドの拡張
IOS SLB FWLB の相互運用（IP リバーススティッキ）

表 1-2　CSM フィーチャ セットの説明（続き）

機能
同一シャーシに複数の CSM
同一シャーシでの CSM および IOS-SLB 機能の同時使用
設定可能な HTTP 1.1 の連続機能（同一サーバにすべての GET が作成される、または複数のサーバにバランシングされる）
全面的に設定可能な NAT
サーバ開始型接続
ルート ヘルス導入
ロードバランシング アルゴリズム
ラウンドロビン
Weighted Round-Robin（WRR; 重み付きラウンドロビン）
実サーバに対してスロースタートできるようにする最小接続機能
重み付き最小接続
URL ハッシュ
送信元 IP ハッシュ（設定可能なマスク）
宛先 IP ハッシュ（設定可能なマスク）
送信元および宛先 IP ハッシュ（設定可能なマスク）
サポート対象ロードバランシング
SLB（TCP、UDP、または総称 IP プロトコル）
ファイアウォール ロードバランシング
DNS ロードバランシング
ステルス ファイアウォール ロードバランシング
トランスペアレント キャッシュ リダイレクト
リバース プロキシ キャッシュ
SSL オフロード
VPN-IPsec ロードバランシング
一般的な IP 装置とプロトコル
スティッキ性
設定可能なオフセットおよび長さを持つ Cookie sticky
SSL ID
送信元 IP（設定可能なマスク）
HTTP リダイレクト
冗長性
sticky ステート
完全ステートフル フェールオーバー（接続の冗長性）

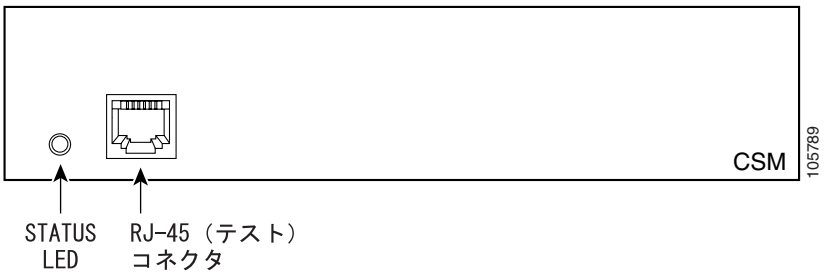
表 1-2　CSM フィーチャ セットの説明（続き）

機能
ヘルス チェック
HTTP
ICMP
Telnet
TCP
FTP
SMTP
DNS
戻りエラー コード チェック
帯域内ヘルス チェック
ユーザ定義による TCL スクリプト
管理
SNMP トラップ
SNMP および MIB フルサポート
リモートの CSM 設定用の XML インターフェイス
バックエンド暗号化のサポート
ワークグループ マネージャのサポート
Server Application State Protocol（SASP）

前面パネル

図 1-1 に、CSM の前面パネルを示します。

図 1-1　CSM の前面パネル



（注） RJ-45 コネクタは着脱式プレートで覆われています。

STATUS LED

CSM の電源が入ると、各種ハードウェア コンポーネントが初期化され、スーパバイザ エンジンとの通信が行われます。STATUS LED は、スーパバイザ エンジンの動作と初期化の結果を示します。通常の初期化シーケンスの間に、STATUS LED は消灯状態からレッド、オレンジ、グリーンへと変化します。

（注） スーパバイザ エンジンの LED の詳細については、『*Catalyst 6500 Series Switch Module Installation Guide*』を参照してください。

表 1-3 に、STATUS LED の動作を示します。

表 1-3　CSM の STATUS LED

色	説明
消灯	• モジュールはスーパバイザ エンジンからの電力供給を待機しています。 • モジュールはオンラインではありません。 • モジュールに電力が供給されていません。次の原因が考えられます。 — CSM に電力が供給されていない。 — モジュール温度が制限値を超えている[1]。
レッド	• スーパバイザ エンジンによるリセットでモジュールが解放され、起動中です。 • ブート コードの実行に失敗した場合、LED は起動後もレッドのままです。

表 1-3　CSM の STATUS LED（続き）

色	説明
オレンジ	• モジュールがハードウェアを初期化中、またはスーパバイザ エンジンと通信中です。 • 初期化シーケンス中にエラーが発生しました。 • モジュールは起動時に FPGA[2] をダウンロードできませんでしたが、初期化シーケンスを続行し、スーパバイザ エンジンからモジュール オンライン ステータスを得ます。 • モジュールはスーパバイザ エンジンからモジュール オンライン ステータスを得ていません。この問題は、CSM に発行された外部ループバック テストでスーパバイザ エンジンがエラーを検出した場合に発生します。
グリーン	• モジュールは動作可能です。スーパバイザ　エンジンからモジュールにモジュール オンライン ステータスが与えられています。
グリーンからオレンジ	• スーパバイザ エンジンの CLI[3] で **set module disable** *mod* コマンドを使用した結果、モジュールがディセーブルになっています。

1. CSM の 4 つの各センサーの温度を表示するには、**show environment temperature** *mod* コマンドを入力します。
2. FPGA = Field Programmable Gate Arrays
3. CLI = コマンドライン インターフェイス

RJ-45 コネクタ

着脱式プレートで覆われた RJ-45 コネクタを使用して、管理ステーションまたはテスト装置を接続します。このコネクタはフィールド エンジニアがテストを行ったり、ダンプ情報を取得したりするために使用します。

CSM の動作

特定の VLAN の設定の場合、クライアントおよびサーバは、レイヤ 2 およびレイヤ 3 テクノロジーを使用して、CSM を介して通信します（図 1-2 を参照）。単純な SLB では、クライアントはクライアント側 VLAN に、サーバはサーバ側 VLAN に接続します。サーバおよびクライアントは異なるサブネット上に配置できます。レイヤ 3 ホップで 1 つまたは複数離れた位置にサーバを配置し、ルータを介して CSM に接続することもできます。

クライアントはモジュールの VIP アドレスのいずれかに要求を送信します。CSM はこの要求を応じることのできるサーバに転送します。サーバはさらに、CSM に応答を転送し、CSM がクライアントにその応答を転送します。

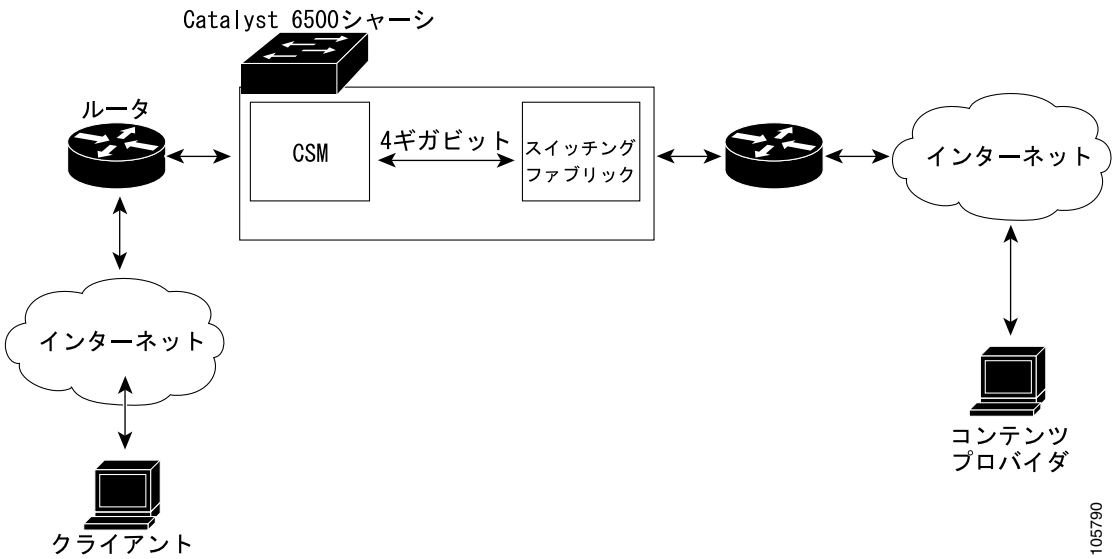
クライアント側およびサーバ側 VLAN が同じサブネット上にある場合は、CSM をシングル サブネット（ブリッジ）モードで設定することができます。詳細については、「シングル サブネット（ブリッジ）モードの設定」（p.2-2）を参照してください。

クライアント側およびサーバ側 VLAN が異なるサブネット上にある場合は、セキュア（ルータ）モードで動作するように CSM を設定できます。詳細については、「セキュア（ルータ）モードの設定」（p.2-4）を参照してください。

冗長 CSM を使用して、セキュア（ルータ）モードまたはシングル サブネット（ブリッジ）モードのどちらでもフォールトトレラント構成を設定できます。詳細については、「フォールトトレランスの設定」（p.7-2）を参照してください。

複数の VLAN を使用して、シングル サブネット（ブリッジ）モードおよびセキュア（ルータ）モードを同じ CSM で共存させることができます。

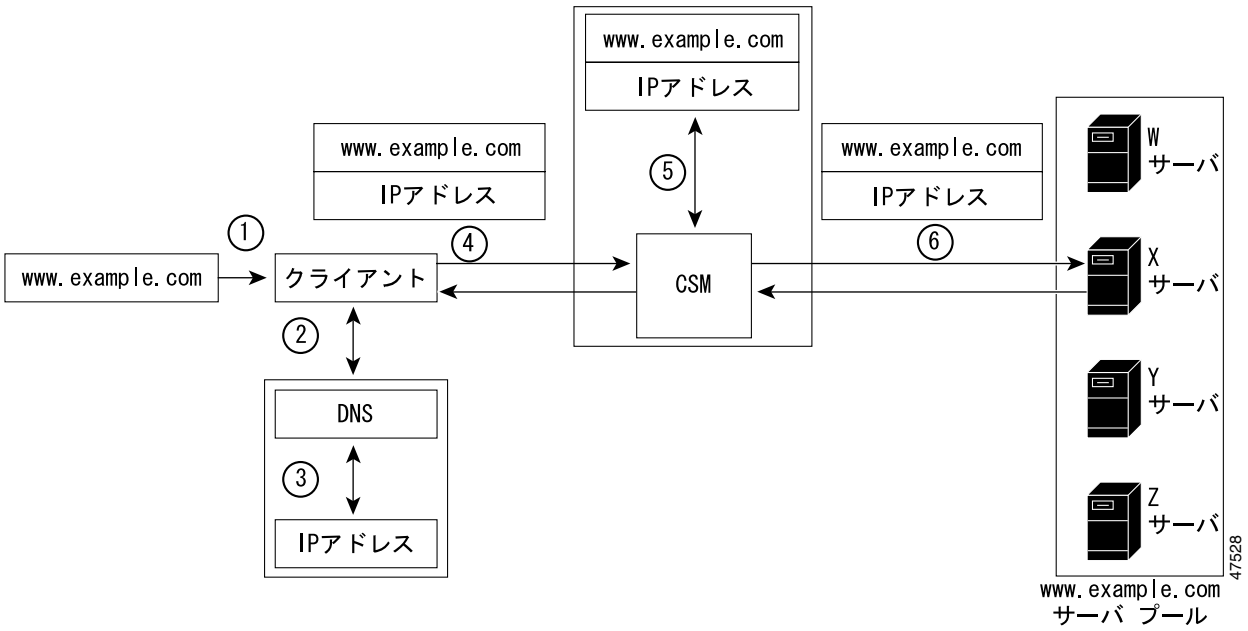
図 1-2　CSM およびサーバ



CSM のトラフィック フロー

ここでは、CSM 環境のクライアントとサーバ間でトラフィックが送信される仕組みについて説明します。（図 1-3 を参照）。

図 1-3　クライアントとサーバ間のトラフィック フロー



（注）　図 1-3 の番号は、次の手順の番号と対応しています。

URL を入力して情報を要求した場合、トラフィック フローは次のようになります。

1. URL を入力します（図 1-3 では、例として www.example.com と示されています）。
2. クライアントは DNS サーバにアクセスして、URL に関連付けられている IP アドレスを検索します。
3. DNS サーバは VIP の IP アドレスをクライアントに送信します。
4. クライアントはその IP アドレス（CSM VIP）を使用して、HTTP 要求を CSM に送信します。
5. CSM は URL と要求を受信し、ロードバランス上の決定を行い、サーバを選択します。

   たとえば、図 1-3 では、CSM は www.example.com サーバ プールからサーバ（X サーバ）を選択し、その VIP アドレスを X サーバのアドレスで置き換えて（directed モード）、トラフィックを X サーバに転送します。NAT サーバ オプションがディセーブルの場合、VIP アドレスは変わりません（dispatch モード）。
6. CSM は NAT を実行し、最終的に TCP シーケンス番号変換を行います。

CHAPTER 2

CSM によるネットワーキング

この章では、Content Switching Module（CSM; コンテント スイッチング モジュール）のネットワーキングについて説明します。

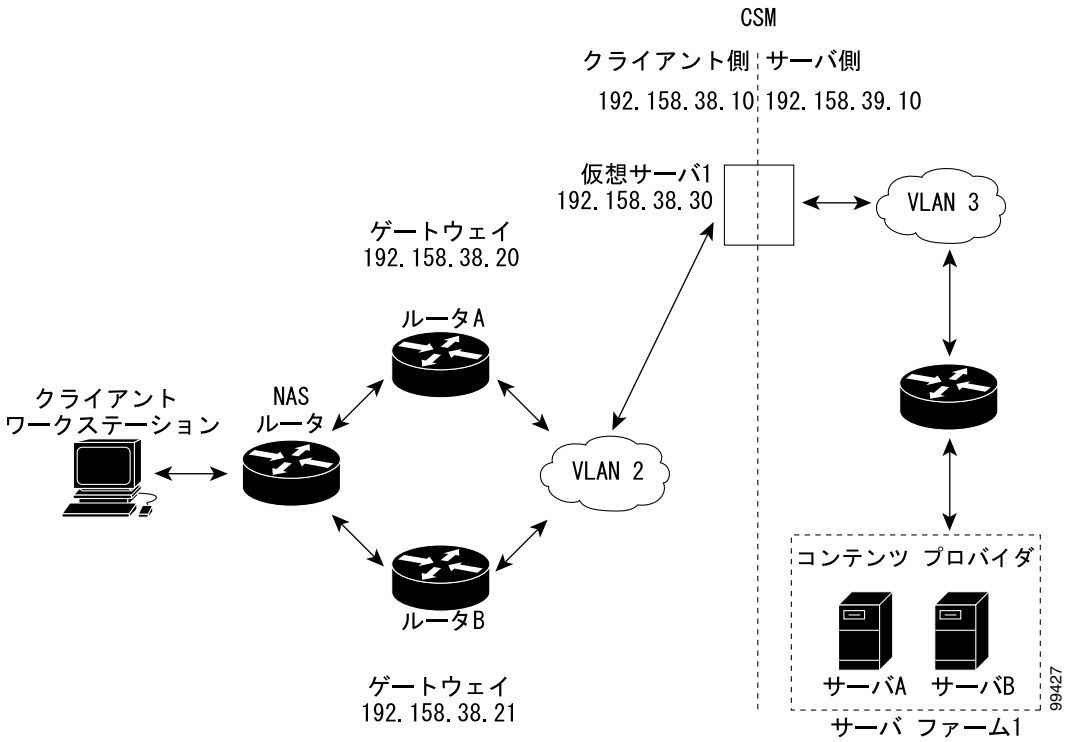
ネットワーキング用モードの設定

シングル サブネット（ブリッジ）モードおよびセキュア（ルータ）モードで CSM を設定できます。ここでは、モードについて説明します。

シングル サブネット（ブリッジ）モードの設定

シングル サブネット（ブリッジ）モード コンフィギュレーションでは、クライアント側およびサーバ側 VLAN（仮想 LAN）を同一サブネット上に配置します。図 2-1 に、シングル サブネット（ブリッジ）モード コンフィギュレーションの設定方法を示します。

図 2-1　シングル サブネット（ブリッジ）モードの設定



（注）　図 2-1 のアドレスは、次の手順の各ステップに関連しています。

（注）　シングル サブネット（ブリッジ）モードを設定するには、CSM のクライアントおよびサーバ VLAN に同じ IP アドレスを割り当てます。

コンテント スイッチングをシングル サブネット（ブリッジ）モードとして設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# vlan database`	VLAN モードを開始します[1]。
ステップ 2	`Router(vlan)# vlan 2`	クライアント側 VLAN を設定します[2]。
ステップ 3	`Router(vlan)# vlan 3`	サーバ側 VLAN を設定します。
ステップ 4	`Router(vlan)# exit`	モードを終了して、設定を有効にします。
ステップ 5	`Router(config-module-csm)# vlan 2 client`	クライアント側 VLAN 2 を作成し、SLB VLAN モードを開始します[1]。
ステップ 6	`Router(config-slb-vlan-client)# ip addr 192.158.38.10 255.255.255.0`	VLAN 2 に CSM の IP アドレスを割り当てます。
ステップ 7	`Router(config-slb-vlan-client)# gateway 192.158.38.20`	ルータ A へのクライアント側 VLAN ゲートウェイを定義します。
ステップ 8	`Router(config-slb-vlan-client)# gateway 192.158.38.21`	ルータ B へのクライアント側 VLAN ゲートウェイを定義します。
ステップ 9	`Router(config-slb-vserver)# vlan 3 server`	サーバ側 VLAN 3 を作成し、SLB VLAN モードを開始します。
ステップ 10	`Router(config-slb-vlan-client)# ip addr 192.158.38.10 255.255.255.0`	VLAN 3 に CSM の IP アドレスを割り当てます。
ステップ 11	`Router(config-slb-vlan-client)# exit`	サブモードを終了します。
ステップ 12	`Router(config-module-csm)# vserver VIP1`	仮想サーバを作成し、SLB 仮想サーバ モードを開始します。
ステップ 13	`Router(config-slb-vserver)# virtual 192.158.38.30 tcp www`	仮想 IP アドレスを作成します。
ステップ 14	`Router(config-slb-vserver)# serverfarm farm1`	仮想サーバをサーバ ファームに関連付けます[3]。
ステップ 15	`Router(config-module-csm)# inservice`	サーバをイネーブルにします。

1. モードまたはサブモードを終了するには、**exit** コマンドを入力します。メニューのトップ レベルに戻るには、**end** コマンドを入力します。
2. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。
3. このステップでは、サーバ ファームが設定してあるものと想定しています（「サーバ ファームの設定」[p.5-2] を参照）。

（注） サーバのデフォルト ルートをルータ A のゲートウェイ（192.158.38.20）またはルータ B のゲートウェイ（192.158.38.21）に設定します。

セキュア（ルータ）モードの設定

セキュア（ルータ）モードでは、クライアント側およびサーバ側 VLAN は異なるサブネット上にあります。

コンテント スイッチングをセキュア（ルータ）モードに設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# vlan database`	VLAN モードを開始します[1]。
ステップ 2	`Router(vlan)# vlan 2`	クライアント側 VLAN を設定します[2]。
ステップ 3	`Router(vlan)# vlan 3`	サーバ側 VLAN を設定します。
ステップ 4	`Router(vlan)# exit`	モードを終了して、設定を有効にします。
ステップ 5	`Router(config-module-csm)# vlan 2 client`	クライアント側 VLAN 2 を作成し、SLB VLAN モードを開始します。
ステップ 6	`Router(config-slb-vlan-client)# ip addr 192.158.38.10 255.255.255.0`	VLAN 2 に CSM の IP アドレスを割り当てます。
ステップ 7	`Router(config-slb-vlan-client)# gateway 192.158.38.20`	ルータ A へのクライアント側 VLAN ゲートウェイを定義します。
ステップ 8	`Router(config-slb-vlan-client)# gateway 192.158.38.21`	ルータ B へのクライアント側 VLAN ゲートウェイを定義します。
ステップ 9	`Router(config-module-csm)# vlan 3 server`	サーバ側 VLAN 3 を作成し、SLB VLAN モードを開始します。
ステップ 10	`Router(config-slb-vlan-server)# ip addr 192.158.39.10 255.255.255.0`	VLAN 3 に CSM の IP アドレスを割り当てます。
ステップ 11	`Router(config-slb-vlan-server)# exit`	サブモードを終了します。
ステップ 12	`Router(config-module-csm)# vserver VIP1`	仮想サーバを作成し、SLB 仮想サーバ モードを開始します。
ステップ 13	`Router(config-slb-vserver)# virtual 192.158.38.30 tcp www`	仮想 IP アドレスを作成します。
ステップ 14	`Router(config-slb-vserver)# serverfarm farm1`	仮想サーバをサーバ ファームに関連付けます[3]。
ステップ 15	`Router(config-module-csm)# inservice`	サーバをイネーブルにします。

1. モードまたはサブモードを終了するには、**exit** コマンドを入力します。メニューのトップ レベルに戻るには、**end** コマンドを入力します。
2. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。
3. このステップでは、サーバ ファームが設定してあるものと想定しています（「サーバ ファームの設定」[p.5-2] を参照）。

（注） サーバのデフォルト ルートを CSM の IP アドレス（192.158.39.10）に設定します。

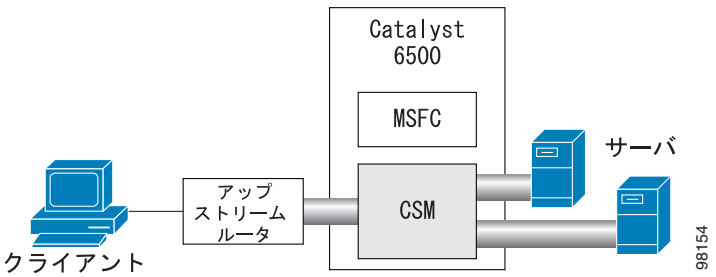
CSM のネットワーキング トポロジー

ここでは、CSM のネットワーキング トポロジーについて説明します。

CSM はインラインで、MSFC は関連しない場合

図 2-2 に、Multilayer Switch Feature Card（MSFC; マルチレイヤ スイッチ フィーチャ カード）との相互運用性がないレイヤ 3 構成での CSM を示します。

図 2-2　CSM はインラインで、MSFC は関連しない場合



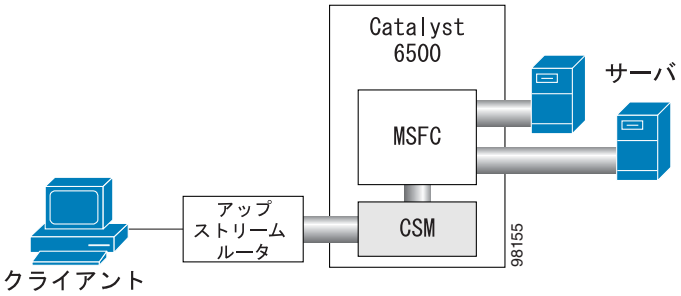
この構成には、次の特性があります。

- MSFC は、CSM VLAN（仮想 LAN）をルーティングしていません。
- サーバ間通信（ダイレクト レイヤ 3 に、またはロードバランスされた）はすべて、CSM を経由します。
- CSM は、アップストリーム ルータ（デフォルト ゲートウェイ）にスタティック ルートを使用する必要があります。

CSM はインラインで、MSFC はサーバ側にある場合

図 2-3 に、MSFC がサーバ側にある構成での CSM を示します。

図 2-3　CSM はインラインで、MSFC はクライアント側にある場合



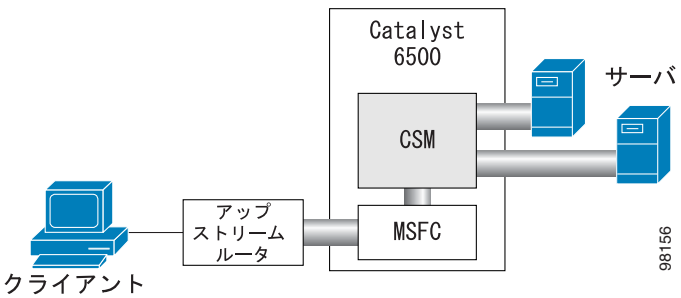
この構成には、次の特性があります。

- サーバ間のダイレクト通信は、CSM をすり抜けます。
- サーバ間のロードバランスされた接続には、常に Secure NAT（SNAT）が必要です。
- CSM は、アップストリーム ルータ（デフォルト ゲートウェイ）にスタティック ルートを使用する必要があります。
- ルーティング プロトコルは、バック エンドで使用可能です。
- レイヤ 2 リライトは、不可能です。

CSM はインラインで、MSFC はクライアント側にある場合

図 2-4 に、MSFC がクライアント側にある構成での CSM を示します。

図 2-4　CSM はインラインで、MSFC はクライアント側にある場合



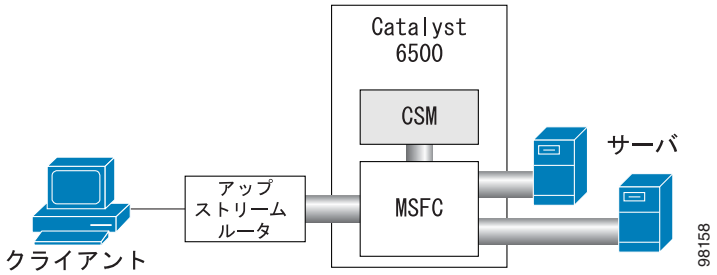
この構成には、次の特性があります。

- この構成は、配置が容易です。
- サーバ間のレイヤ 3 通信は、CSM を通過します。
- ルーティング プロトコルは、MSFC とアップストリーム ルータ間で使用可能です。
- サーバとの間のトラフィックはすべて、CSM を通過します。

集約モードの CSM

図 2-5 に、集約モード構成での CSM を示します。

図 2-5　集約モードの CSM



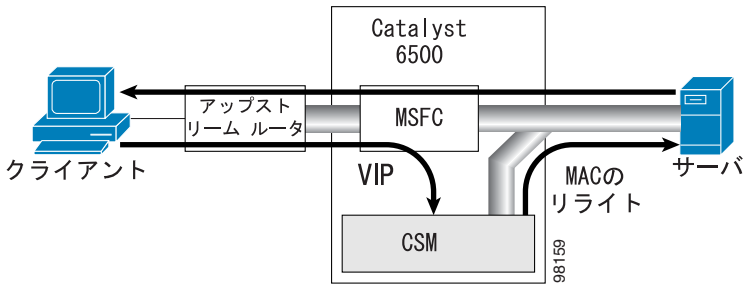
この構成には、次の特性があります。

- CSM がインラインでないため、モジュールは不必要なトラフィックを認識しません。
- ルーティングと CSM 構成が容易です。
- 戻りトラフィックが必要なので、Policy Based Routing（PBR; ポリシー ベース ルーティング）またはクライアント SNAT が必要です。
- サーバ間のロードバランシングされた接続には、常に SNAT が必要です。
- レイヤ 2 リライトは、不可能です。

ダイレクト サーバ リターン

図 2-6 に、ダイレクト サーバ リターン構成の CSM を示します。

図 2-6　ダイレクト サーバ リターン



この構成には、次の特性があります。

- ロードバランサで、高度なスループットまたは帯域幅が必要ありません。
- ロードバランサは、戻りトラフィックを認識しません。
- TCP フローは、常にタイムアウトである必要があります。
- TCP 終端が不可能です（レイヤ 4 ロードバランシングのみ）。
- 帯域内ヘルス モニタリングが不可能です。
- サーバはレイヤ 2 に隣接して、ループバック アドレスを持つ必要があります。

CSM のルーティング

ロードバランス接続を転送および維持する際は、CSM がルーティング決定をする必要があります。ただし CSM は、ルーティング プロトコルの実行、および MSFC ルーティング テーブルへのアクセスを行いません。CSM は 3 タイプのエントリを使用して、CSM 独自のルーティング テーブルを構築します。

- 直接接続される IP サブネット

  これらのサブネットは、CSM クライアントまたはサーバ VLAN で設定されます。

- デフォルト ゲートウェイ

  デフォルト ゲートウェイは、クライアントまたはサーバ VLAN コンフィギュレーション サブモード内部から、**gateway** キーワードを使用して設定されます。第 4 章「VLAN の設定」を参照してください。このリリースでは、最大 511 のデフォルト ゲートウェイを設定できます。ただし、同一 VLAN には、7 つ以上のデフォルト ゲートウェイを設定することができません。

  ほとんどの構成では、デフォルト ゲートウェイは（または簡略化されて）1 つです。このゲートウェイは、アップストリーム ルータ（またはアップストリーム ルータ ペアを表す HSRP IP アドレス）を示し、結果的に、多様なスタティック ルートを示しています。

- スタティック ルート

  スタティック ルートは、クライアントまたはサーバ VLAN コンフィギュレーション サブモード内部から、**route** キーワードを使用して設定されます。第 4 章「VLAN の設定」を参照してください。スタティック ルートは、サーバがレイヤ 2 に隣接していない場合に便利です。

複数のデフォルト ゲートウェイがサポートされます。ただし、CSM が未知の宛先にルーティング決定をする必要がある場合、CSM は干渉または制御されることなく、無作為に 1 つのゲートウェイを選択します。この動作を制御するには、後に説明するプレディクタ転送オプションを使用します。

CSM は、次の 3 つの状況のときにルーティング決定をする必要があります。

- 新たな接続を受信した場合。

  CSM は、この時点でその接続の戻りトラフィックの送信先を決定する必要があります。他の装置とは異なり、CSM はルート ルックアップを実行しませんが、接続の最初のパケットを受信した場所に基づいて送信元 MAC アドレスを記憶します。この接続用の戻りトラフィックは、送信元 MAC アドレスに送り返されます。また、この動作はアップストリーム ルータ間の冗長プロトコル（HSRP など）と連動します。

- CSM がルータ モードで設定されている場合。

  サーバは CSM をデフォルト ゲートウェイとして示して、接続を開始します。

- サーバ ファームが、プレディクタ転送オプションで設定されている場合（第 5 章「実サーバおよびサーバ ファームの設定」を参照）。このプレディクタは CSM に接続のロードバランシングではなく、ルーティングを指示します。

ゲートウェイが複数ある場合、固有の実サーバとしてのゲートウェイで設定されたサーバ ファームを使用して、最初の 2 つの状況を簡略化することができます。「サーバを送信元とする VIP への接続用の送信元 NAT の設定」（p.A-8）を参照してください。

DoS 攻撃からの保護

CSM は、ロードバランシングする装置の保護、および CSM 自体を DoS 攻撃から保護するために、さまざまな機能を実装しています。これらの機能の多くは、CSM により制御され着信トラフィック量に適応しているため、ユーザが設定することはできません。

CSM は、次の DoS 保護機能を提供します。

- SYN Cookie

  （注） SYN Cookie は Cookie の同期化とは異なる機能なので、混同しないでください。ここでは、SYN Cookie についてのみ言及します。

  保留接続数が設定可能なスレッシュホールドを超過すると、CSM は SYN Cookie を使用して、生成されたシーケンス番号ですべての接続のステート情報を暗号化し始めます。その結果、CSM は TCP 接続の保留（完全に確立されていない）にフロー ステートを消費できないようになります。この動作はハードウェアに完全実装され、SYN 攻撃からの保護に役立っています。

- 接続保留タイムアウト

  この機能は仮想サーバ単位で設定可能で、秒単位で指定された設定タイムアウト時間内に適切に確立されていない接続をタイムアウトにすることができます。

- 接続アイドル タイムアウト

  この機能は仮想サーバ単位で設定可能で、確立された接続が、タイマーで設定されたインターバル時間内にトラフィックを渡さなかった場合、タイムアウトにすることができます。

- 一般 TCP 終端

  レイヤ 7 ロードバランシングに TCP 終端を必要としない接続もあります。これらの実サーバとの接続をロードバランシングする前に、すべての着信 TCP 接続を終了するよう任意の仮想サーバに設定することができます。この設定により、レイヤ 4 ロードバランシング環境にあるすべての CSM DoS 機能を利用できるようになります。

CHAPTER 3

設定前の作業

この章では、Content Switching Module（CSM; コンテント スイッチング モジュール）の設定を始める前の必要事項について説明します。

オペレーティング システムのサポート

CSM は、スーパバイザ エンジン上で Catalyst オペレーティング システムを稼働し、Multilayer Switch Feature Card（MSFC; マルチレイヤ スイッチ フィーチャ カード）上で Cisco IOS を稼働するスイッチ上でサポートされます。また、CSM はスーパバイザ エンジンと MSFC の両方で Cisco IOS ソフトウェアを稼働するスイッチでもサポートされます。

CSM は MSFC CLI で設定されるため、Catalyst オペレーティング システムと Cisco IOS ソフトウェアの両方を稼働するスイッチを使用している場合は、MSFC への最初のセッションで、CSM を設定するための MSFC CLI へアクセスする必要があります。MSFC CLI へアクセスすると、Catalyst オペレーティング システムのスイッチの場合でも Cisco IOS スイッチの場合でも CSM の設定は同じになります。

Catalyst オペレーティング システムと Cisco IOS ソフトウェアの両方が稼働するスイッチを使用する場合は、レイヤ 2 の設定（VLAN およびポート アソシエーションなど）はすべて、スーパバイザ エンジン上で実行されます。

（注） Cisco IOS ソフトウェアのみを稼働するスイッチ上で CSM を実行すると、設定された VLAN（仮想 LAN）が CSM とスイッチのバックプレーンを接続するトランクまたはチャネルに自動的に追加されます。Catalyst オペレーティング システムと Cisco IOS ソフトウェアの両方が稼働するスイッチでは、CSM VLAN を手動でトランクまたはチャネルに追加する必要があります。

CSM の設定準備

CSM を設定する前に、次の作業を終えておく必要があります。

- スイッチとモジュールの Cisco IOS バージョンが一致していることを確認します。『*Catalyst 6500 Series Switch Content Switching Module Installation Guide*』を参照してください。
- Server Load Balancing（SLB; サーバ ロードバランシング）を設定するには、次の情報を入手しておく必要があります。
  - 導入先で使用するネットワーク トポロジー
  - 実サーバの IP アドレス
  - Domain Name Server（DNS; ドメイン ネーム サーバ）で使用する CSM VIP 用エントリ（名前を使用して DNS にアクセスする必要がある場合）
  - 各仮想サーバの IP アドレス
- 先に Catalyst 6500 シリーズ スイッチで VLAN を設定してから、CSM に対して VLAN を設定する必要があります。スイッチとモジュールの VLAN ID は同じでなければなりません。詳細については、『*Catalyst 6500 Series Switch Software Configuration Guide*』を参照してください。

  次に、VLAN を設定する例を示します。

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# vlan 130
Router(config-vlan)# name CLIENT_VLAN
Router(config-vlan)# exit
Router(config)# vlan 150
Router(config-vlan)# name SERVER_VLAN
Router(config-vlan)# end
```

- サーバまたはクライアントに接続する物理インターフェイスを、対応する VLAN 内に組み込みます。

  次に、物理インターフェイスをレイヤ 2 インターフェイスとして設定して、VLAN に割り当てる例を示します。

```
Router>
Router> enable
Router# config
Router(config)# interface 3/1
Router(config-if)# switchport
Router(config-if)# switchport access vlan 150
Router(config-if)# no shutdown
Router(vlan)# exit
```

- クライアント側またはサーバ側 VLAN 上のネクストホップ ルータで MSFC を使用する場合は、対応するレイヤ 3 VLAN インターフェイスを設定する必要があります。

注意　ポリシー ベース ルーティングまたは送信元 Network Address Translation（NAT; ネットワーク アドレス変換）を使用し、CSM をルータ モードとして設定しないかぎり、クライアント側とサーバ側の両方に対して、MSFC をルータとして同時に使用することはできません。このような状況が発生するのは、CSM がロードバランスまたは転送する両方向のフローを確認する必要があるためです。ブリッジ（シングル サブネット）モードで CSM を使用する場合は、クライアント側とサーバ側の両方に対して、MSFC 上でレイヤ 3 VLAN インターフェイスを設定しないでください。CSM をルータ モードで使用する場合は、戻りトラフィックが CSM に送り返されるように、ポリシー ベース ルーティングまたは送信元 NAT を適切に設定しないかぎり、クライアント側とサーバ側の両方に対して、MSFC 上でレイヤ 3 VLAN インターフェイスを設定しないでください。

次に、レイヤ 3 VLAN インターフェイスを設定する例を示します。

```
Router>
Router> enable
Router# config
Router(config)# interface vlan 130
Router(config-if)# ip address 10.10.1.10 255.255.255.0
Router(config-if)# no shutdown
Router(vlan)# exit
```

コマンドライン インターフェイスの使用法

CSM のソフトウェア インターフェイスは Cisco IOS CLI です。Cisco IOS CLI および Cisco IOS コマンド モードの詳細については、『*Catalyst 6500 Series Switch Cisco IOS Software Configuration Guide*』の Chapter 2 を参照してください。

（注） プロンプトに入力できる文字数に制限があるため、プロンプトが切り捨てられる場合があります。たとえば、
Router(config-slb-vlan-server)# は Router(config-slb-vlan-serve)# のように表示されます。

オンライン ヘルプの利用方法

どのコマンド モードでも、疑問符（?）を入力すると、使用できるコマンドのリストが表示されます。

```
Router> ?
```

または

```
Router(config)# module csm 5
Router(config-module-csm)# ?
```

（注） オンライン　ヘルプでは、コマンドで使用できるデフォルトのコンフィギュレーション値、および値の範囲が表示されます。

設定の保存および復元

設定の保存および復元については、『*Catalyst 6500 Series Switch Cisco IOS Software Configuration Guide*』を参照してください。

SLB モードの設定

Catalyst 6500 シリーズ スイッチの SLB 機能は、2 種類のモードで設定されます。Routed Processor（RP）モードおよび CSM モードです。スイッチの設定は、CSM の動作に影響しません。CSM はデフォルトで、RP モードで設定されます。RP モードを使用すると、同一シャーシで 1 つまたは複数の CSM を設定し、同一スイッチで Cisco IOS SLB を実行できます。

（注）　RP モードはデフォルト モードで、推奨されているモードです。CSM モードは、Release 2.1 より前の CSM ソフトウェア イメージとの下位互換性を維持するためだけに使用されます。新しい CSM または CSM イメージを搭載する場合は、RP モードを使用してください。

CSM モードの場合、設定できるのは 1 つの CSM だけです。CSM モードをサポートしているのは、旧ソフトウェア リリースとの下位互換性を維持するためです。単一 CSM 設定では、同一スイッチで Cisco IOS SLB を実行できません。

次に、モードについて説明します。

モードのコマンド構文

スイッチに CSM コンフィギュレーション コマンドを入力するには、その前に設定対象の CSM を指定しなければなりません。設定する CSM を指定するには、**module csm** *slot-number* コマンドを使用します。*slot-number* 値は、設定対象の CSM が搭載されているシャーシ スロットです。

module csm コマンドによって CSM コンフィギュレーション サブモードが開始されます。以後、入力したすべてのコンフィギュレーション コマンドは、指定したスロットに搭載された CSM に適用されます。

（注）　特に明記していないかぎり、このマニュアルの例はすべて、このコマンドが入力済みで、設定対象の CSM に対応するコンフィギュレーション サブモードがすでに開始されていることが前提です。

CSM モードおよび RP モードを設定するコマンドの構文は、次の点を除いて同じです。

- CSM モードで設定する場合、上位レベルのコマンドごとに、プレフィクスとして **ip slb** を指定する必要があります。
- CSM モードの設定と RP モードの設定では、プロンプトが異なります。

スロット 5 の CSM に仮想サーバを設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config)# module csm 5`	設定する CSM の位置を指定します。
ステップ 2	`Router(config-module-csm)# vserver VS1`	仮想サーバを設定します。

次に、config-module-csm モードでのすべてのコマンドのリスト例を表示します。

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# module csm 5
Router(config-module-csm)# ?
SLB CSM module config
  arp          configure a static ARP entry
  capp         configure Content Application Peering Protocol
  default      Set a command to its defaults
  dfp          configure Dynamic Feedback Protocol manager
  exit         exit SLB CSM module submode
  ft           configure CSM fault tolerance (ft) feature
  map          configure an SLB map
  natpool      configure client nat pool
  no           Negate a command or set its defaults
  owner        configure server owner
  policy       configure an SLB policy
  probe        configure an SLB probe
  real         configure module real server
  script       configure script files and tasks
  serverfarm   configure a SLB server farm
  standby
  static       configure static NAT for server initiated connections
  sticky       configure a sticky group
  variable     configure an environment variable
  vlan         configure a vlan
  vserver      configure an SLB virtual server
  xml-config   settings for configuration via XML
```

モード間の切り替え

ip slb mode コマンドを使用してモードを CSM から RP に切り替えると、既存の CSM 設定が新しい設定に移行します。既存の CSM 設定がある場合は、スロット番号を要求されます。

Catalyst 6500 シリーズ スイッチでは、RP モードの設定から CSM モードの設定に移行できます。Cisco IOS SLB の設定から CSM 設定への移行は、手動にかぎって可能です。

CSM モードと RP モードの相違

CSM モードと RP モードは、CSM が CLI から設定される方法に影響するだけで、CSM 自体の動作および機能に影響するわけではありません。CSM と同一シャーシ内の Cisco IOS SLB の場合と同様に、1 つのシャーシ内の複数の CSM を設定する場合は、RP モードである必要があります。

CSM モード

ip slb mode csm コマンド モードを使用して、1.x リリースの CSM を設定することができます。このモードでは、シャーシ内の単一の CSM を設定できます（同じシャーシ内でその他の CSM または Cisco IOS SLB を設定することはできません）。

このモードでは、すべての CSM コンフィギュレーション コマンドは **ip slb** で始まります。

CSM の **show** コマンドは、**show ip slb** で始まります。

CSM 2.1 以降のリリースを使用している場合は、このモードは推奨できません。このモードは、Cisco IOS CLI で下位互換性の維持のためにオプションとして提供されているからです。

次に、シャーシ内の単一の CSM の設定例を示します。

```
Cat6k# show running-config
Building configuration...
Current configuration : 5617 bytes

ip slb mode csm
ip slb vlan 110 server
ip address 10.10.110.1 255.255.255.0

ip slb vlan 111 client
ip address 10.10.111.2 255.255.255.0
gateway 10.10.111.1

ip slb probe HTTP_TEST http
request method get url /probe/http_probe.html
expect status 200
interval 5
failed 5

ip slb serverfarm WEBFARM
nat server
no nat client
real 10.10.110.10
inservice
real 10.10.110.20
inservice
probe HTTP_TEST

ip slb vserver HTTPVIP
virtual 10.10.111.100 tcp www
persistent rebalance
serverfarm WEBFARM
inservice
```

RP モード

Cisco IOS SLB が稼働するシャーシで複数の CSM を設定するには、**ip slb mode rp** コマンド モード（デフォルト）を使用します。このモードで設定できるのは、Release 2.1 以降の CSM だけです。

このモードでは、CSM は次のコマンド サブモードから設定されます。

mod csm *X*

X は設定する CSM のスロット番号です。

CSM の show コマンドは、**show mod csm** *X* で始まります。

CSM ソフトウェア Release 2.1 以降で CSM を設定する場合は、RP モードが推奨されます。このモードでも、Cisco IOS SLB に適用されるすべてのコマンドは、シャーシ内の CSM には適用されません。これらのコマンドは、**ip slb** で始まります。

次に、シャーシ内の単一の CSM の設定例を示します。

```
Cat6k# show running-config
Building configuration...

Current configuration : 5597 bytes
!---

module ContentSwitchingModule 5
vlan 110 server
ip address 10.10.110.1 255.255.255.0

vlan 111 client
ip address 10.10.111.2 255.255.255.0
gateway 10.10.111.1

probe HTTP_TEST http
request method get url /probe/http_probe.html
expect status 200
interval 5
failed 5

serverfarm WEBFARM
nat server
no nat client
real 10.10.110.10
inservice
real 10.10.110.20
inservice
probe HTTP_TEST

vserver HTTPVIP
virtual 10.10.111.100 tcp www
persistent rebalance
serverfarm WEBFARM
inservice
```

モードの変更

CSM の動作モードを CSM モードから RP モードに、または RP モードから CSM モードに変更することができます。次に、モードの変更方法を示します。

CSM モードから RP モード

CSM モードから RP モードに変更する例を示します。これは CSM 1.x から 2.1 以降のリリースへの一般的な移行例で、モジュールのリセットは必要ありません。

```
Cat6k# configure terminal
 Enter configuration commands, one per line.  End with CNTL/Z.

Cat6k(config)# ip slb mode ?
   csm  SLB in Content Switching Module
   rp   SLB in IOS system

Cat6k(config)# ip slb mode rp
 % The current SLB mode is CSM-SLB.
 % You are selecting RP-SLB mode.
 % All configuration for CSM-SLB will be moved to module submode.
 % Confirm switch to RP-SLB mode? [no]: yes
 % Enter slot number for CSM module configuration, 0 for none [5]: 5
 % Please save the configuration.
Cat6k(config)# end

Cat6k# write
 Building configuration...
 [OK]
Cat6k#
```

RP モードから CSM モード

これは RP モードから CSM モードへの移行例で、モジュールのリセットが必要です。

```
Cat6k# configure terminal
 Enter configuration commands, one per line.  End with CNTL/Z.

Cat6k(config)# ip slb mode ?
   csm  SLB in Content Switching Module
   rp   SLB in IOS system

Cat6k(config)# ip slb mode csm
 % The current SLB mode is RP-SLB.
 % You are selecting CSM-SLB.
 % All SLB configurations for RP will be ERASED.
 % After execution of this command, you must
 % write the configuration to memory and reload.
 % CSM-SLB module configuration will be moved to ip slb submodes.
 % Confirm switch to CSM-SLB mode? [no]: yes
 % Enter slot number for CSM module configuration, 0 for none [5]: 5
 % Please save the configuration and reload.

Cat6k(config)# end
Cat6k# write
 Building configuration...
Cat6k# reload
 Proceed with reload? [confirm] y
Verify Mode Operation
```

設定の確認

設定が適切に機能するかどうかを確認するには、RP モードで次のコマンドを使用します。

```
Cat6k# show ip slb mode
     SLB configured mode = rp

Cat6k# configure terminal
 Enter configuration commands, one per line.  End with CNTL/Z.

Catk6-1(config)# ip slb ?
   dfp           configure Dynamic Feedback Protocol manager
   entries       initial and maximum SLB entries
   firewallfarm  configure an SLB firewall farm
   mode          configure SLB system mode
   natpool       define client nat pool
   probe         configure an SLB probe
   serverfarm    configure an SLB server farm
   vserver       configure an SLB virtual server
```

設定が適切に機能するかどうかを確認するには、Cisco IOS SLB モードで次のコマンドを使用します。

```
Cat6k(config)# module csm 5
Cat6k(config-module-csm)# ?
 SLB CSM module config
   default     Set a command to its defaults
   dfp         configure Dynamic Feedback Protocol manager
   exit        exit SLB CSM module submode
   ft          configure CSM fault tolerance (ft) feature
   map         configure an SLB map
   natpool     configure client nat pool
   no          Negate a command or set its defaults
   policy      configure an SLB policy
   probe       configure an SLB probe
   serverfarm  configure an SLB server farm
   static      configure static NAT for server initiated connections
   sticky      configure a sticky group
   vlan        configure a vlan
   vserver     configure an SLB virtual server
```

シャーシ内の単一の CSM 設定が適切に機能するかどうかを確認するには、CSM モードで次のコマンドを使用します。

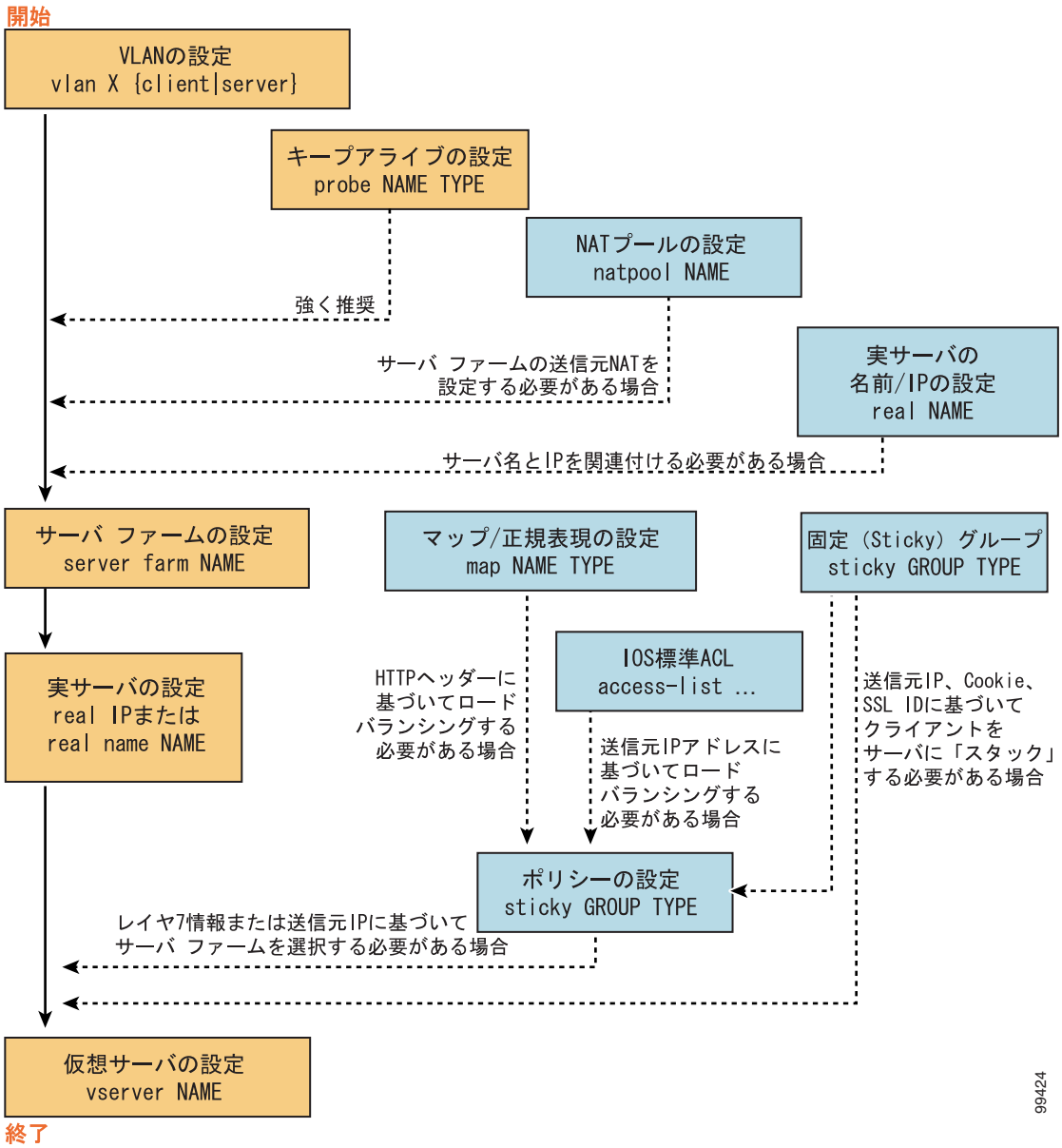
```
Cat6k# show ip slb mode
     SLB configured mode = csm

Catk6-1# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.

Cat6k(config)# ip slb ?
   dfp         configure Dynamic Feedback Protocol manager
   ft          configure CSM fault tolerance (ft) feature
   map         configure an SLB map
   mode        configure SLB system mode
   natpool     configure client nat pool
   policy      configure an SLB policy
   probe       configure an SLB probe
   serverfarm  configure an SLB server farm
   static      configure static NAT for server initiated connections
   sticky      configure a sticky group
   vlan        configure a vlan
   vserver     configure an SLB virtual server
```

設定の概要

ここで説明する設定プロセスは、スイッチが RP モードであることを前提としています。図 3-1 は、設定プロセスで必須の操作と任意の操作の概要を示しています。

（注）　レイヤ 4 ロードバランシングの場合、ポリシーの設定は不要です。

図 3-1　設定の概要



必要なパラメータの設定については、次の各項を参照してください。

- クライアント側 VLAN の設定（p.4-3）
- サーバ側 VLAN の設定（p.4-4）
- サーバ ファームの設定（p.5-2）
- 実サーバの設定（p.5-4）
- 仮想サーバの設定（p.6-2）

必須のロードバランシング パラメータを CSM 上で設定すると、任意のパラメータを設定することができます。次の各項を参照してください。

- リダイレクト仮想サーバの設定（p.6-8）
- クライアント NAT プールの設定（p.5-8）
- サーバ開始型接続の設定（p.5-9）
- TCP パラメータの設定（p.6-5）

高度な設定と連動するには、第 2 章から第 11 章の次の項目を参照してください。

- シングル サブネット（ブリッジ）モードの設定（p.2-2）
- セキュア（ルータ）モードの設定（p.2-4）
- URL ハッシュの設定（p.5-10）
- 一般ヘッダー解析の設定（p.6-14）
- RHI の設定（p.8-7）
- フォールトトレランスの設定（p.7-2）
- 連続（persistent）接続の設定（p.8-19）
- HSRP の設定（p.7-6）
- 接続の冗長性の設定（p.7-12）
- 実サーバの SNMP トラップの設定（p.8-26）
- ヘルス モニタリング用プローブの設定（p.9-2）
- 帯域内ヘルス モニタリングの設定（p.9-10）
- HTTP 戻りコード チェックの設定（p.9-11）
- CSM での TCL スクリプトの使用（p.10-1）
- ステルス ファイアウォール ロードバランシングの設定（p.11-8）
- 標準ファイアウォール ロードバランシングの設定（p.11-18）
- ファイアウォール用リバーススティッキの設定（p.11-27）

新しいソフトウェア リリースへのアップグレード

ここでは、CSM をアップグレードする 3 種類の方法について説明します。

- スーパバイザ エンジン ブートフラッシュからのアップグレード（p.3-12）
- PCMCIA カードからのアップグレード（p.3-13）
- 外部 TFTP サーバからのアップグレード（p.3-14）

（注） 新しいソフトウェア リリースにアップグレードする場合は、CSM イメージをアップグレードしてから Cisco IOS イメージをアップグレードする必要があります。そうしないと、スーパバイザ エンジンが CSM を認識しません。スーパバイザ エンジンが CSM を認識しない場合は、Cisco IOS イメージをダウングレードし、CSM イメージをアップグレードしてから、Cisco IOS イメージをアップグレードする必要があります。

CSM をアップグレードするには、アップグレードする CSM モジュールとのセッションを開始する必要があります。アップグレード中は、スーパバイザ エンジンに接続されたコンソールにすべてのコマンドを入力します。コンフィギュレーション コマンドは、それぞれ異なるコマンドラインに入力してください。アップグレードを完了するには、**exit** コマンドを入力して、スーパバイザ エンジン プロンプトに戻ります。「SLB モードの設定」（p.3-4）を参照してください。

注意 アップグレードしている CSM とのセッションを終了するには、**exit** コマンドを入力する必要があります。セッションを終了しないで CSM を Catalyst 6500 シリーズ シャーシから取り外した場合は、CSM にコンフィギュレーション コマンドを入力するために、**Ctrl** + **^** を押して、**x** を入力し、プロンプトに **disconnect** コマンドを入力する必要があります。

スーパバイザ エンジン ブートフラッシュからのアップグレード

（注） ブートフラッシュにイメージをロードする手順については、『*Catalyst 6500 Series Supervisor Engine Flash PC Card Installation Note*』を参照してください。

スーパバイザ エンジンのブートフラッシュから CSM をアップグレードする手順は、次のとおりです。

ステップ 1　TFTP サーバをイネーブルにして、次のようにブートフラッシュからイメージを供給します。

```
Router>
Router> enable
Router# configure terminal
Router(config)# tftp-server sup-bootflash:c6slb-apc.revision-num.bin
Router(config)
```

ステップ 2　スーパバイザ エンジンと CSM 間のセッションを確立します。

```
Router# session slot csm-slot-number processor 0
```

ステップ 3　スーパバイザ エンジンから CSM にイメージをロードします。

```
CSM> upgrade 127.0.0.zz c6slb-apc.revision-num.bin
```

シャーシ スロット 1 に搭載されたスーパバイザ エンジンの場合、*zz* は 12 です。
シャーシ スロット 2 に到来されたスーパバイザ エンジンの場合、*zz* は 22 です。

（注）　スーパバイザ エンジンを搭載できるのは、シャーシのスロット1またはスロット2だけです。

ステップ 4　CSM とのセッションを終了し、Cisco IOS プロンプトに戻ります。

```
CSM> exit
```

ステップ 5　CSM の電源を切断して再投入するか、またはスーパバイザ エンジンのコンソールから次のコマンドを入力して、CSM を再起動します。

```
Router(config)# hw-module module csm-slot-number reset
```

PCMCIA カードからのアップグレード

（注）　このマニュアルでは全体を通して、*PCMCIA カード*の代わりに*フラッシュ PC カード*という言葉を使用します。

スーパバイザ エンジンに装着された着脱式フラッシュ PC カードから CSM をアップグレードする手順は、次のとおりです。

ステップ 1　TFTP サーバをイネーブルにして、着脱式フラッシュ PC カードからイメージを供給します。

```
Router>
Router> enable
Router# configure terminal
Router(config)# tftp-server slotx:c6slb-apc.revision-num.bin
```

*x*値＝0（フラッシュPCカードがスーパバイザ エンジンのPCMCIAスロット0に搭載されている場合）

ステップ 2　スーパバイザ エンジンと CSM 間のセッションを確立します。

```
Router# session slot csm-slot-number processor 0
```

ステップ 3　スーパバイザ エンジンから CSM にイメージをロードします。

```
CSM> upgrade slot0: c6slb-apc.revision-num.bin
```

（注）　スーパバイザ エンジンを搭載できるのはシャーシのスロット 1 またはスロット 2 だけです。

ステップ 4　CSM とのセッションを終了し、Cisco IOS プロンプトに戻ります。

```
CSM> exit
```

ステップ 5　CSM の電源を切断して再投入するか、またはスーパバイザ エンジンのコンソールから次のコマンドを入力して、CSM を再起動します。

```
Router# hw-module module csm-slot-number reset
```

外部 TFTP サーバからのアップグレード

外部 TFTP サーバから CSM をアップグレードする手順は、次のとおりです。

ステップ 1　TFTP CSM ランタイム イメージ ダウンロード用の VLAN をスーパバイザ エンジン上に作成します。

（注）　既存の VLAN も使用できますが、確実にダウンロードするには、TFTP 接続専用の VLAN を作成する必要があります。

ステップ 2　TFTP サーバに接続するインターフェイスを設定します。

ステップ 3　インターフェイスを VLAN に追加します。

ステップ 4　CSM の **vlan** コマンドを入力します。

詳細については、第 4 章「VLAN の設定」を参照してください。

ステップ 5　CSM 用 VLAN に IP アドレスを追加します。

ステップ 6　**show csm** *slot* **vlan detail** コマンドを入力し、設定を確認します。

詳細については、第 4 章「VLAN の設定」を参照してください。

ステップ 7　CSM から TFTP サーバに接続できるかどうかを確認します。

```
Router# ping module csm csm-slot-number TFTP-server-IP-address
```

ステップ 8　スーパバイザ エンジンと CSM 間のセッションを確立します。

```
Router# session slot csm-slot-number processor 0
```

ステップ 9　イメージをアップグレードします。

```
CSM> upgrade TFTP-server-IP-address c6slb-apc.rev-number.bin
```

ステップ 10　CSM とのセッションを終了し、Cisco IOS プロンプトに戻ります。

```
CSM> exit
```

ステップ 11　CSM の電源を切断して再投入するか、またはスーパバイザ エンジンのコンソールから次のコマンドを入力して、CSM を再起動します。

```
Router# hw-module module csm-slot-number reset
```

CHAPTER 4

VLAN の設定

この章では、Content Switching Module（CSM; コンテント スイッチング モジュール）上に VLAN（仮想 LAN）を設定する方法について説明します。

Catalyst 6500 シリーズ スイッチに CSM を搭載した場合、クライアント側およびサーバ側 VLAN を設定する必要があります（図 4-1 を参照）。

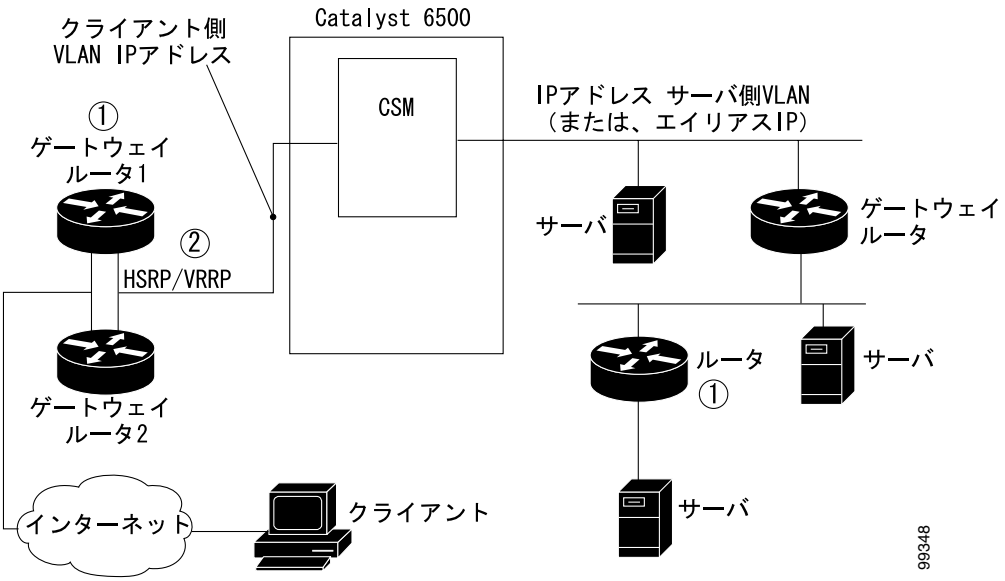
クライアント側またはサーバ側 VLAN という専門用語は、クライアント側を向いている VLAN とサーバまたは宛先装置に接続している VLAN を論理的に区別しています。ただし、CSM のクライアントおよびサーバ VLAN の機能は非常に類似しています。たとえば、新しい接続はサーバ側 VLAN で受信され、次にクライアント側 VLAN に向けてロードバランスされます。

クライアント側 VLAN とサーバ側 VLAN の違いは、次のとおりです。

- ブリッジ モードを設定する場合、2 つのサーバ VLAN または 2 つのクライアント VLAN をブリッジすることはできません。1 つのクライアント VLAN と 1 つのサーバ VLAN しかブリッジできません。
- DoS の保護機能は、特にレート制限制御トラフィックが CPU から送信される場合などは、クライアント側 VLAN の方が積極的です。

（注） 先に Catalyst 6500 シリーズ スイッチ上で VLAN を設定してから、CSM に対して VLAN を設定する必要があります。スイッチとモジュールの VLAN ID は同じでなければなりません。

図 4-1　VLAN の設定



図の注記：

1 ― CSM はトラフィックを転送するためのレイヤ 3 ルックアップを実行しません。したがって、CSM は ICMP リダイレクトに応答することができません。

2 ― VLAN ごとに最大 7 つのゲートウェイを設定できます。システム全体でのクライアントおよびサーバ VLAN の最大数は 511、ゲートウェイの最大数は 224 です。Hot Standby Router Protocol（HSRP）ゲートウェイを設定した場合、CSM は 224 のゲートウェイ エントリのうちの 3 つを使用します。トラフィックが HSRP グループの仮想および物理 MAC アドレスから届く可能性があるからです（「HSRP の設定」[p.7-6] を参照）。フォールトトレラント VLAN は IP インターフェイスを使用しないので、512 の VLAN 制限には適用されません。

クライアント側 VLAN の設定

クライアント側 VLAN を設定する手順は、次のとおりです。

注意　VLAN 1 を CSM のクライアント側またはサーバ側 VLAN として使用することはできません。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# vlan vlanid client`	クライアント側 VLAN を設定し、クライアント VLAN モードを開始します[1]。
ステップ 2	`Router(config-slb-vlan-client)# ip active_ip_addr [netmask] [alt standby_ip_addr [netmask]]`	この VLAN 上のプローブおよび ARP 要求で使用される IP アドレスをアクティブな CSM に設定します。冗長 CSM を使用している場合は、**alt** キーワードを入力して、スタンバイ CSM へ送信される代替 IP アドレスを指定します。[2]
ステップ 3	`Router(config-slb-vlan-client)# description description`	（任意）VLAN の説明を指定します。説明は最大 80 文字までです。
ステップ 4	`Router(config-slb-vlan-client)# gateway ip-address`	ゲートウェイ IP アドレスを設定します。

1. モードまたはサブモードを終了するには、**exit** コマンドを入力します。メニューのトップ レベルに戻るには、**end** コマンドを入力します。
2. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。

次に、CSM をクライアント側 VLAN 用に設定する例を示します。

```
Router(config-module-csm)# vlan 130 client
Router(config-slb-vlan-client)# ip addr 123.44.50.6 255.255.255.0 alt 123.44.50.7
255.255.255.0
Router(config-slb-vlan-client)# gateway 123.44.50.1
Router(config-slb-vlan-client)# exit
```

サーバ側 VLAN の設定

サーバ側 VLAN を設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# vlan vlanid server`	サーバ側 VLAN を設定し、サーバ VLAN モードを開始します[1]。
ステップ 2	`Router(config-slb-vlan-server)# ip active_ip_addr [netmask] [alt standby_ip_addr [netmask]]`	サーバ側 VLAN 用に IP アドレスを設定します。冗長 CSM を使用している場合は、**alt** キーワードを入力して、スタンバイ CSM へ送信される代替 IP アドレスを指定します。[2]
ステップ 3	`Router(config-slb-vlan-server)# description description`	（任意）VLAN の説明を指定します。説明は最大 80 文字までです。
ステップ 4	`Router(config-slb-vlan-server)# alias ip-address netmask`	（任意）実サーバの代替ゲートウェイとして、複数の IP アドレスを CSM に設定します[3]。
ステップ 5	`Router(config-slb-vlan-server)# route ip-address netmask gateway gw-ip-address`	CSM から実サーバまでのレイヤ 3 ホップ数が 2 以上の場合は、実サーバまでのスタティック ルートを設定します。
ステップ 6	`Router # show module csm slot vlan [client | server | ft] [id vlan-id] [detail]`	クライアント側およびサーバ側 VLAN 設定を表示します。

1. モードまたはサブモードを終了するには、**exit** コマンドを入力します。メニューのトップ レベルに戻るには、**end** コマンドを入力します。
2. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。
3. 冗長構成ではエイリアスが必要です。第 7 章「冗長性の設定」を参照してください。

次に、CSM をサーバ側 VLAN 用に設定する例を示します。

```
Router(config-module-csm)# vlan 150 server
Router(config-slb-vlan-server)# ip addr 123.46.50.6 255.255.255.0
Router(config-slb-vlan-server)# alias 123.60.7.6 255.255.255.0
Router(config-slb-vlan-server)# route 123.50.0.0 255.255.0.0 gateway 123.44.50.1
Router(config-slb-vlan-server)# exit
```

CHAPTER 5

実サーバおよびサーバ ファームの設定

この章では、サーバおよびサーバ ファームの設定方法について説明します。

サーバ ファームの設定

サーバ ファームまたはサーバ プールは、同じコンテンツが含まれるサーバの集合です。サーバ ファームを設定してサーバを追加するとき、およびサーバ ファームを仮想サーバにバインドするときは、サーバ ファーム名を指定します。サーバ ファームを設定する手順は、次のとおりです。

- サーバ ファーム名を指定します。
- ロードバランシング アルゴリズム（プレディクタ）およびファームのその他の属性を設定します。
- 1 組の実サーバを設定または指定します（「実サーバの設定」[p.5-4] を参照）。
- 実サーバの属性を設定または指定します。

各サーバ ファームに帯域内ヘルス モニタリングを設定することもできます（「帯域内ヘルス モニタリングの設定」[p.9-10] を参照）。サーバ ファームに戻りコード マップを割り当てると、戻りコードの解析を設定することができます（「HTTP 戻りコード チェックの設定」[p.9-11] を参照）。

サーバ ファームを設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# serverfarm serverfarm-name`	サーバ ファームを作成し、名前を付けて、サーバ ファーム コンフィギュレーション モードを開始します[1,2]。
ステップ 2	`Router(config-slb-sfarm)# predictor [roundrobin | leastconns [slowstart timer]| hash url | hash address [source | destination] [ip-netmask] | forward]]`	ロードバランシング予測アルゴリズムを設定します[2]。指定しなかった場合のデフォルトは **roundrobin** です。 **slowstart** *timer* コマンドは、スロースタート メカニズムが期限切れになるまでのタイマーを設定します。*timer* の有効な値は 1 ～ 65535 秒です。**slowstart** *timer* はデフォルトではディセーブルです。
ステップ 3	`Router(config-slb-sfarm)# nat client client-pool-name`	（任意）NAT モード クライアントをイネーブルにします[2]（「クライアント NAT プールの設定」[p.5-8] を参照）。 **（注）** サーバファームとポリシーの両方にクライアント NAT が設定されている場合は、ポリシーがサーバファームより優先されます。
ステップ 4	`Router(config-slb-sfarm)# no nat server`	（任意）ロードバランシング決定時に宛先 IP アドレスを変更しないことを指定します。
ステップ 5	`Router(config-slb-sfarm)# probe probe-name`	（任意）**probe** コマンドを使用して定義できるプローブにサーバ ファームを関連付けます[2]。
ステップ 6	`Router(config-slb-sfarm)# bindid bind-id`	（任意）1 つの物理サーバを複数のサーバ ファームにバインドして、それぞれに異なるウェイトを伝えます[2]。**bindid** コマンドは Dynamic Feedback Protocol（DFP）で使用されます。
ステップ 7	`Router(config-slb-sfarm)# failaction {purge | reassign}`	（任意）実サーバとの接続が失敗した場合の動作を設定します[2]。

	コマンド	目的
ステップ 8	`Router(config-slb-sfarm)# description description`	（任意）サーバ ファームの説明を指定します。説明は最大 80 文字までです。
ステップ 9	`Router(config-slb-sfarm)# health retries 20 failed 600`	サーバ ファーム内のすべてのサーバに帯域内ヘルス モニタリングを設定します。
ステップ 10	`Cat6k-2(config-slb-sfarm)# retcode-map NAME_OF_MAP`	HTTP 戻りエラー コード チェックを設定します（タイプ レコードのマップの設定が必要です）。
ステップ 11	`Router(config-slb-sfarm)# real ip_address`	実サーバを定義します。
ステップ 12	`Router(config-slb-real)# inservice`	実サーバをイネーブルにします。
ステップ 13	`Router# show module csm slot serverfarm serverfarm-name [detail]`	特定またはすべてのサーバ ファームに関する情報を表示します。

1. モードまたはサブモードを終了するには、**exit** コマンドを入力します。メニューのトップ レベルに戻るには、**end** コマンドを入力します。
2. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。

最小接続プレディクタを設定する場合は、スロースタート メカニズムを実行して、高いレートの新しい接続をサービス状態になったばかりのサーバに送信しないようにします。アクティブ接続数が最も少ない実サーバに、最小接続プレディクタによる次のサーバ ファーム接続要求が与えられます。

REAL_SLOW_START_ENABLE の環境変数は、実サーバがサービス状態になるときの起動レートを制御します。スロースタート起動は、「最小接続」方式が設定されたサーバ ファームだけを対象とします。

この変数に設定できる範囲は、0 ～ 10 です。0 を設定すると、スロースタート機能がディセーブルになります。1 ～ 10 の値は、新たにアクティブになったサーバが起動する速度を指定します。1 の値は、最も遅い起動レートです。10 の値は、Content Switching Module（CSM; コンテント スイッチング モジュール）が新たにアクティブ化されたサーバにより多くの要求を割り当てるように指定します。3 の値は、デフォルト値です。

設定値が N の場合、CSM は最初から 2 ^ N（2 の N 乗）の新規の要求を新たにアクティブになったサーバに割り当てます（その際、接続が終端されていないことを前提とします）。このサーバがさらに多くの接続を終了、または終端すると、より早い起動が行われます。新たにアクティブにされたサーバの現在のオープン接続数がサーバ ファームのほかのサーバと同じになるか、スロースタート タイマーが期限切れになると、起動が停止します。

次に、最小接続（**leastconns**）アルゴリズムを使用して p1_nat という名前のサーバ ファームを設定する例を示します。

```
Router(config-module-csm)# serverfarm p1_nat
Router(config-slb-sfarm)# description Server Farm Example
Router(config-slb-sfarm)# predictor leastconns
Router(config-slb-sfarm)# real 10.1.0.105
Router(config-slb-real)# inservice
Router(config-slb-sfarm)# real 10.1.0.106
Router(config-slb-real)# inservice
```

実サーバの設定

実サーバはサーバ ファームに割り当てられた物理装置です。実サーバはロードバランス対象のサービスを提供します。クライアント要求を受信したサーバは、CSM に応答を送信し、CSM からクライアントにその応答が転送されます。

実サーバ コンフィギュレーション モードで実サーバを設定するには、実サーバをサーバ ファームに割り当てるときにサーバの IP アドレスおよびポートを指定します。サーバ ファーム モードから実サーバ コンフィギュレーション モードを開始して、実サーバを追加します。

実サーバは次のように設定できます。

- no inservice ― CSM サービスを実行しません。適用される固定（sticky）および新規の接続はありません。

（注） no inservice を指定した場合、CSM はオープンしている接続を削除しません。オープンしている接続を削除するには、**clear module csm** *slot* **connection** コマンドを使用して手動でこの操作を実行する必要があります。

- inservice ― CSM サービスを実行します。モジュールに対して固定（sticky）および新規接続を設定できます。
- inservice standby ― CSM サービスをスタンバイ状態にします。固定（sticky）を設定できます。新規接続は設定できません。

実サーバを設定するには、次の作業を実行します。

	コマンド	目的
ステップ 1	`Router(config-slb-sfarm)# real` *ip-address* [*port*]	実サーバをサーバ ファームのメンバーとして指定し、実サーバ コンフィギュレーション モードを開始します。オプションの変換ポートを設定することもできます[1,2]。
ステップ 2	`Router(config-slb-real)# weight` *weighting-value*	（任意）ラウンドロビンまたは最小接続が選択されている場合は、仮想サーバ予測アルゴリズムの重み付け値を設定して、サーバ ファーム内のその他のサーバに対する相対的なサーバの作業負荷容量を割り当てます[2]。 **（注）** サーバの順序が先頭（最初のサーバ）から始められるのは、設定時またはサーバ ステートの変更（プローブまたは DFP エージェントのいずれか）時のみです。 最小接続プレディクタを設定する場合は、スロースタート メカニズムを実行して、高いレートの新しい接続をサービス状態になったばかりのサーバに送信しないようにします。
ステップ 3	`Router(config-slb-real)# maxconns` *max-conns*	（任意）実サーバのアクティブな接続数の最大値を設定します[2]。アクティブな接続数が指定した最大値に到達すると、接続数が最小スレッシュホールドより小さくなるまで、新しい接続が実サーバに送信されなくなります。

	コマンド	目的
ステップ 4	`Router(config-slb-real)# minconns min-conns`	（任意）最小接続スレッシュホールドを設定します[2]。
ステップ 5	`Router(config-slb-real)# inservice`	CSM で使用できるように、実サーバをイネーブルにします[2、3]。
ステップ 6	`Router# show module csm slot [sfarm serverfarm-name] [detail]`	（任意）設定された実サーバに関する情報を表示します。**sfarm** オプションを指定すると、特定の仮想サーバに関連付けられた実サーバに関する情報だけが表示されます。**detail** オプションを指定すると、実サーバの詳細が表示されます。
ステップ 7	`Router# show module csm slot [vserver virtserver-name] [client ip-address] [detail]`	CSM に対するアクティブな接続が表示されます。**vserver** オプションを指定すると、特定の仮想サーバに関連付けられた接続に関する情報だけが表示されます。**client** オプションを指定すると、特定のクライアントの接続だけが表示されます。**detail** オプションを指定すると、接続の詳細が表示されます。

1. モードまたはサブモードを終了するには、**exit** コマンドを入力します。メニューのトップ レベルに戻るには、**end** コマンドを入力します。
2. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。
3. 設定する実サーバごとに、ステップ 1 ～ステップ 5 を繰り返します。

次に、実サーバを作成する例を示します。

```
Router(config-module-csm)# serverfarm serverfarm
Router(config-slb-sfarm)# real 10.8.0.7
Router(config-slb-real)# inservice
Router(config-slb-sfarm)# real 10.8.0.8
Router(config-slb-real)# inservice
Router(config-slb-sfarm)# real 10.8.0.9
Router(config-slb-real)# inservice
Router(config-slb-sfarm)# real 10.8.0.10
Router(config-slb-real)# inservice
Router(config-slb-sfarm)# real 10.1.0.105
Router(config-slb-real)# inservice
Router(config-slb-sfarm)# real 10.1.0.106
Router(config-slb-sfarm)# inservice
Router(config-slb-real)# end
Router# show mod csm slot reals detail
Router# show mod csm slot conns detail
```

no inservice コマンドを使用することによって、実サーバがサービスを停止すると、CSM は適切なサーバ シャットダウンを実行します。このコマンドを使用すると、既存のセッションは完了またはタイム アウトされますが、すべての新しいセッションは実サーバへのロードバランシングが停止されます。新しいセッションは、この仮想サーバ用のサーバ ファーム内のほかのサーバにロードバランシングされます。

（注） no inservice を指定した場合、CSM はオープンしている接続を削除しません。オープンしている接続を削除するには、**clear module csm** *slot* **conn** コマンドを使用して手動でこの操作を実行する必要があります。

スタンバイ状態では、ファイアウォールに障害が発生した場合、失敗したアクションを再度割り当てることができます。ファイアウォール接続の再割り当てを設定するには、次の 3 つの適切なシャットダウン オプションを使用します。

- サーバ ファームに失敗したアクションの再割り当てを設定します。
- アクションの失敗に備えて、ほかの実サーバのバックアップとして実サーバの 1 つを割り当てます。
- バックアップ実サーバは、inservice を有効にするか、またはスタンバイ バックアップ状態に設定します。スタンバイ状態では、プライマリ実サーバに障害が発生した場合にのみ、この実サーバが新規接続を受け付けます。

サービスから実サーバを削除する例を示します。

```
Router(config-slb-real)# no inservice
```

サーバ ファームの設定の詳細については、「サーバ ファームの設定」（p.5-2）を参照してください。

実サーバがヘルス プローブに失敗し、サービスが停止される場合も、CSM は適切なサーバ シャットダウンを実行します。CSM ヘルス プローブの設定の詳細については、「ヘルス モニタリング用プローブの設定」（p.9-2）を参照してください。

要求をするクライアントが使用不能なサーバ（Cookie、SSL ID、送信元 IP などを使用している）に固定されている場合、この接続はファーム内の使用可能なサーバに分散されます。使用不能なサーバに固定される場合は、**inservice standby** コマンドを入力してください。**inservice standby** コマンドを入力すると、このサーバに固定される接続および既存の接続があるサーバを除けば、いかなる接続もスタンバイ実サーバに送信されません。指定のスタンバイ時間のあと、**no inservice** コマンドを使用することにより、既存のセッションだけを実サーバに送信することができます。次に、固定（sticky）接続がサーバ ファーム内のサービス中の実サーバに送信されます。

DFP の設定

DFP を設定すると、サーバから CSM にフィードバックしてロードバランシングを強化することができます。DFP により、（物理サーバ上の）ホスト エージェントは仮想サービスを提供するホスト システムのステータス変化を動的に報告できます。

（注） DFP エージェントは任意のホスト マシンに配置できます。DFP エージェントは、エージェントによって管理される実サーバの IP アドレスおよびポート数には関係しません。DFP マネージャは、DFP エージェントとの接続を確立し、DFP エージェントからロード ベクターを受信します。

DFP を設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# dfp [password password]`	DFP マネージャを設定し、オプションのパスワードを指定し、DFP エージェント サブモードを開始します[1,2]。
ステップ 2	`Router(config-slb-dfp)# agent ip-address port [activity-timeout [retry-count [retry-interval]]]`	キープアライブ メッセージの間隔、連続する接続試行回数または無効な DFP レポート数、および接続試行の間隔を設定します[2]。
ステップ 3	`Router# show module csm slot dfp [agent [detail | ip-address port] | manager [ip_addr] | detail | weights]`	DFP マネージャおよびエージェント情報を表示します。

1. モードまたはサブモードを終了するには、**exit** コマンドを入力します。メニューのトップ レベルに戻るには、**end** コマンドを入力します。
2. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。

次に、DFP を設定する例を示します。

```
Router(config-module-csm)# dfp password password
Router(config-slb-dfp)# agent 123.234.34.55 5 6 10 20
Router(config-slb-dfp)# exit
```

クライアント NAT プールの設定

クライアントの Network Address Translation（NAT; ネットワーク アドレス変換）プールを設定すると、NAT によってクライアント要求の送信元 IP アドレスがサーバ側 VLAN（仮想 LAN）の IP アドレスに変換されます。**nat** コマンドの serverfarm サブモードで NAT プール名を使用して、クライアント NAT プール用として設定すべき接続を指定します。

クライアント NAT プールを設定するには、次の作業を実行します。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# natpool pool-name start-ip end-ip netmask mask`	コンテント スイッチング NAT を設定します。このコマンドを使用するには、1 つまたは複数のクライアント アドレス プールを作成する必要があります[1,2]。
ステップ 2	`Router(config-module-csm)# serverfarm serverfarm-name`	serverfarm サブモードを開始して、クライアント NAT を適用します。
ステップ 3	`Router(config-slb-sfarm)# nat client clientpool-name`	設定されたNATプールをサーバ ファームに関連付けます。
ステップ 4	`Router# show module csm natpool [name pool-name] [detail]`	NAT の設定を表示します。

1. モードまたはサブモードを終了するには、**exit** コマンドを入力します。メニューのトップ レベルに戻るには、**end** コマンドを入力します。
2. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。

次に、クライアント NAT プールを設定する例を示します。

```
Router(config)# natpool pool1 102.36.445.2 102.36.16.8 netmask 255.255.255.0
Router(config)# serverfarm farm1
Router(config-slb-sfarm)# nat client pool1
```

HTTP のヘッダー挿入は、CSM にクライアント IP アドレスの HTTP ヘッダーへの挿入のような情報を挿入させる機能です。HTTP ヘッダーの挿入はヘッダー マップ内で設定します。設定の詳細については、「HTTP ヘッダー挿入」（p.8-20）を参照してください。

サーバ開始型接続の設定

サーバ用の NAT を使用すると、実サーバで接続を開始することができます。また、サーバ NAT コンフィギュレーション内に一致するエントリがない接続を開始する場合は、サーバに対してデフォルト設定を使用することができます。デフォルトでは、NAT を使用しなくても CSM はサーバを送信元とする接続を確立できます。

サーバ用 NAT を設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config)# static [drop | nat [ip-address | virtual]]`	サーバを送信元とする接続を設定します。オプションにより、接続のドロップ、指定された IP アドレスによる NAT 設定、接続に関連付けられた仮想 IP アドレスによる NAT 設定を行うこともできます[1,2]。
ステップ 2	`Router(config-slb-static)# real ip-address [subnet-mask]`	スタティック NAT サブモードを設定し、サーバにこの NAT オプションを指定します。複数の NAT コンフィギュレーション オプションで同じ実サーバを使用することはできません。

1. モードまたはサブモードを終了するには、**exit** コマンドを入力します。メニューのトップ レベルに戻るには、**end** コマンドを入力します。
2. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。

URL ハッシュの設定

接続用のサーバ ファームを選択すると、そのサーバ ファーム内の特定の実サーバを選択できます。最小接続、ラウンドロビン、または URL ハッシュのいずれかで、実サーバを選択できます。

URL ハッシュは、レイヤ 7 接続に対応するロードバランシング プレディクタです。サーバ ファーム単位で、CSM に URL ハッシュを設定できます。CSM は URL に基づいたハッシュ値を使用して実サーバを選択します。このハッシュ値は、URL 全体で計算する場合と URL の一部を使用して計算する場合があります。URL の一部分でハッシュする場合、URL の中で先頭パターンと終了パターンを指定し、指定された先頭パターンから終了パターンまでの URL 部分だけがハッシュされるようにします。CSM はソフトウェア Release 2.1(1) から URL ハッシュをサポートしています。

先頭パターンと終了パターンを指定しなかった場合（「先頭および終了パターンの設定」[p.5-11] を参照）、URL 全体がハッシュされ、実サーバの選択に使用されます。

URL ハッシュ プレディクタの設定

URL 全体を使用するのか、それとも先頭および終了パターンを使用するのかに関係なく、URL ハッシュ プレディクタを使用する予定のすべてのサーバ ファームについて、URL ハッシュを設定する必要があります。

サーバ ファームにロードバランシング プレディクタとして URL ハッシュを設定する手順は、次のとおりです。

コマンド	目的
`Router(config-slb-sfarm)# predictor hash url`	サーバ ファームに URL ハッシュおよびロードバランシング プレディクタを設定します。

次に、サーバ ファームに URL ハッシュおよびロードバランシング プレディクタを設定する例を示します。

```
Router(config)# mod csm 2
Router(config-module-csm)# serverfarm farm1
Router(config-slb-sfarm)# predictor hash url
Router(config-slb-sfarm)# real 10.1.0.105
Router(config-slb-real)# inservice
Router(config-slb-real)# exit
```

URL ハッシュを使用すると、キャッシュ サーバの機能が向上します。ただし、ハッシュ メソッドは実サーバ用のウェイトを認識しません。実サーバに割り当てられたウェイトは、ラウンドロビンおよび最小接続プレディクタ メソッドで使用されます。

（注） サーバの順序が先頭（最初のサーバ）から始められるのは、設定時またはサーバ ステートの変更（プローブまたは DFP エージェントのいずれか）時のみです。

実サーバに異なるウェイトを作成するのに、サーバ ファーム内のキャッシュ サーバの複数の IP アドレスを表示できます。また、同じ IP アドレスを異なるポート番号で使用することもできます。

（注） サーバ ウェイトはハッシュ プレディクタには使用されません。

URL ハッシュ プレディクタを使用して、実サーバをウェイトで設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-slb-sfarm)# serverfarm MYFARM`	MYFARM という名前のサーバを作成します。
ステップ 2	`Router(config-slb-sfarm)# real 1.1.1.1 80`	ポート 80 の実サーバを指定します。
ステップ 3	`Router(config-slb-sfarm)# inservice`	サービス状態の実サーバをイネーブルにします。
ステップ 4	`Router(config-slb-sfarm)# real 1.1.1.1 8080`	ポート 8080 の実サーバを指定します。
ステップ 5	`Router(config-slb-sfarm)# inservice`	サービス状態の実サーバをイネーブルにします。

先頭および終了パターンの設定

仮想サーバ レベルで先頭パターンおよび終了パターンを設定することができます。定義したパターンは、URL ハッシュがイネーブルに設定されているその仮想サーバのすべてのポリシーに割り当てられたあらゆるサーバ ファームに適用されます。

先頭および終了パターンによって、ハッシュ対象になる URL 部分が区切られ、その仮想サーバに割り当てられたポリシーに属するサーバ ファームから実サーバを選択するためのプレディクタとして使用されます。

URL 全体ではなく、URL の一部分をハッシュする場合は、**vserver vserver-name** サブモードで **url-hash begin-pattern** *pattern-a* コマンドおよび **url-hash end-pattern** *pattern-b* コマンドを使用して、先頭パターンおよび終了パターンを指定します。先頭パターンから終了パターンまでのハッシュが行われます。

たとえば、次の URL で先頭パターンを **c&k=** に、終了パターンを **&** にした場合は、**c&k=c** の部分だけがハッシュされます。

http://quote.yahoo.com/q?s=csco&d=**c&k=c1**&t=2y&a=v&p=s&l=on&z=m&q=l\

（注） 先頭および終了パターンは、固定定数文字列に限定されます。汎用正規表現をパターンとして指定することはできません。先頭パターンを指定しなかった場合、URL の先頭からハッシュが開始されます。終了パターンを指定しなかった場合、URL の末尾でハッシュが終了します。

次に、URL ハッシュの先頭および終了パターンを設定する例を示します。

```
Router(config-module-csm)#
Router(config-module-csm)# vserver vs1
Router(config-slb-vserver)# virtual 10.1.0.81 tcp 80
Router(config-slb-vserver)# url-hash begin-pattern c&k= end-pattern &
Router(config-slb-vserver)# serverfarm farm1
Router(config-slb-vserver)# inservice
Router(config-slb-vserver)#
Router(config-slb-vserver)# exit
Router(config-module-csm)# exit
```

CHAPTER 6

仮想サーバ、マップ、およびポリシーの設定

この章では、コンテント スイッチングの設定方法について説明します。

仮想サーバの設定

ここでは、仮想サーバの設定方法について説明します。

（注）　仮想サーバを IP アドレスで設定すると、このサーバがサービス停止中であっても、この特定の IP に対する Address Resolution Protocol（ARP）要求には応答を開始します。この機能は、特に動作可能な仮想サーバを既存のデバイスから Content Switching Module（CSM; コンテント スイッチング モジュール）に切り替える場合に重要となります。同一ネットワーク内のほかの装置と同じ IP で設定された CSM 上に仮想サーバがないことを確認してください。

仮想サーバは、ポリシーによって実サーバ ファームに関連付けられた、一連の実サーバを表します。仮想サーバを設定するには、デフォルトのサーバ ファーム（デフォルトのポリシー）を指定する仮想サーバの属性を設定し、ポリシー リストを使用して他のサーバ ファームを関連付ける必要があります。デフォルト サーバ ファーム（デフォルト ポリシー）は、要求がどの Server Load Balancing（SLB; サーバ ロード バランシング）ポリシーとも一致しなかった場合、またはポリシーが仮想サーバに関連付けられていない場合に使用されます。

サーバ ファームを仮想サーバに関連付ける前に、サーバ ファームを設定する必要があります。詳細については、「サーバ ファームの設定」（p.5-2）を参照してください。ポリシーは、仮想サーバ コンフィギュレーションに入力された順番で処理されます。詳細については、「ポリシーの設定」（p.6-12）を参照してください。

各仮想サーバに保留接続タイムアウトを設定すると、スイッチがトラフィックで溢れた場合に、接続を迅速に切断することができます。この接続は、要求 / 応答プロセスが完了していないクライアントとサーバ間のトランザクションに適用されます。

カスタマーごとに異なる仮想サーバを割り当てているサービス プロバイダー環境では、CSM の接続リソースの大部分または全部を特定のサーバが吸収しないように接続を分散させる必要があります。

VIP 接続ウォーターマーク機能を使用すると、CSM から特定の仮想サーバに振り分けられる接続数を制限できます。この機能では、仮想サーバごとに限度を設定できるので、全仮想サーバ間で接続リソースが公平に分散されます。

（注）　レベル 4 またはレベル 7 のどちらかで動作する仮想サーバを 1 つだけ設定できます。レベル 4 で動作する仮想サーバを設定する場合は、仮想サーバ コンフィギュレーションの中でサーバ ファーム（デフォルト ポリシー）を指定します（次の手順のステップ 3 を参照）。レベル 7 で動作する仮想サーバを設定する場合は、仮想サーバ コンフィギュレーションに SLB ポリシーを追加します（次の手順のステップ 7 を参照）。

CSM はあらゆる IP プロトコルからのトラフィックに対してロードバランスを図ることができます。仮想サーバ サブモードで仮想サーバを設定するときには、仮想サーバが受け付ける IP プロトコルを定義する必要があります。

（注） すべての IP プロトコルにプロトコル番号がありますが、CSM では対応する番号を入力する代わりに、名前で TCP または UDP を指定できます。

仮想サーバ コンフィギュレーション サブモードで仮想サーバを設定します。

仮想サーバを設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# owner` *owner-name* **address** *street-address-information* **billing-info** *billing-address-information* **email-address** *email-information* **maxconns** *1:MAXULONG*	仮想サーバへのアクセスを特定のオーナー オブジェクトだけに制限します。
ステップ 2	`Router(config-module-csm)# vserver` *virtserver-name*	仮想サーバを特定し、仮想サーバ コンフィギュレーション モードを開始します[1,2]。
ステップ 3	`Router(config-slb-vserver)# vs-owner` *owner-name* **maxconns** *max-conn*	この仮想サーバのオーナー オブジェクト名を設定します。
ステップ 4	`Router(config-slb-vserver)# virtual` *ip-address* [*ip-mask*] *protocol port-number* [**service ftp**]	仮想サーバの任意のポート番号または名前、接続カップリング、および接続タイプに対応する IP アドレスを設定します[2]。*protocol* 値は **tcp**、**udp**、**any**（ポート番号は不要）、または *number* 値（ポート番号は不要）です。
ステップ 5	`Router(config-slb-vserver)# serverfarm` *serverfarm_name*	デフォルト サーバ ファームを仮想サーバに関連付けます[2,3]。指定できるサーバ ファームは 1 つだけです。サーバ ファームを指定しなかった場合、他のあらゆるポリシーと一致しなかった要求は廃棄されます。
ステップ 6	`Router(config-slb-vserver)# sticky` *duration*	（任意）クライアントからの接続が同じ実サーバを使用するように設定します[2、3]。デフォルトでは sticky が無効です。
ステップ 7	`Router(config-slb-vserver)# parse-length` {*bytes* | **default-policy**}	（任意）解析可能な HTTP ヘッダーの最大バイト数を設定します[4]。*bytes* の範囲は 1 ～ 4000 で、デフォルトは 2000 です。デフォルトでは、解析可能な最大長を超えると要求は廃棄されます。 セッションで解析可能な最大長に達したときに、要求を廃棄せずデフォルト ポリシーで処理するように指定するには、**default-policy** キーワードを入力します。
ステップ 8	`Router(config-slb-vserver)# sticky` *group-number* **reverse**	（任意）CSM が該当する方向の接続を同じ送信元に戻すようにします。
ステップ 9	`Router(config-slb-vserver)# client` *ip-address network-mask* [**exclude**]	（任意）仮想サーバを使用できるクライアントを制限します[2、3]。
ステップ 10	`Router(config-slb-vserver)# slb-policy` *policy-name* [**priority** *priority_value*]	（任意）1 つまたは複数のコンテント スイッチング ポリシーを仮想サーバに関連付けます[2]。このポリシーの実行順序を指定するには、**priority** キーワードを入力します。

	コマンド	目的
ステップ 11	`Router(config-slb-vserver)# description` *description*	（任意）仮想サーバの説明を指定します。説明は最大 80 文字までです。
ステップ 12	`Router(config-slb-vserver)# inservice`	CSMで使用できるように、仮想サーバをイネーブルにします[2]。
ステップ 13	`Router# show module csm` *slot* `vserver [details]`	コンテント スイッチング用に定義された仮想サーバの情報を表示します。

1. モードまたはサブモードを終了するには、**exit** コマンドを入力します。メニューのトップ レベルに戻るには、**end** コマンドを入力します。
2. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。
3. これらのパラメータはデフォルトのポリシーを参照します。
4. HTTP ヘッダー全体のバイト数には、既知の URL、すべての Cookie、およびすべてのヘッダー フィールドが含まれます。

次に、barnett という名前の仮想サーバを設定し、bosco という名前のサーバ ファームを関連付けて、固定グループ 12 への 50 分間の固定接続を設定する例を示します。

```
Router(config)# mod csm 2
Router(config-module-csm)# sticky 1 cookie foo timeout 100
Router(config-module-csm)# exit
Router(config-module-csm)#
Router(config-module-csm)# serverfarm bosco
Router(config-slb-sfarm)# real 10.1.0.105
Router(config-slb-real)# inservice
Router(config-slb-real)# exit
Router(config-slb-sfarm)#
Router(config-slb-sfarm)# vserver barnett
Router(config-slb-vserver)# virtual 10.1.0.85 tcp 80
Router(config-slb-vserver)# serverfarm bosco
Router(config-slb-vserver)# sticky 50 group 12
Router(config-slb-vserver)# inservice
Router(config-slb-vserver)# exit
Router(config-module-csm)# end
```

次に、vs1 という名前の仮想サーバを設定し、2 つのポリシーとともに、クライアント トラフィックが特定のポリシーと一致した場合のデフォルト サーバ ファームを指定する例を示します。仮想サーバは、そのポリシーに結合されたサーバ ファームに対して負荷が分散されます。クライアント トラフィックがどのポリシーとも一致しなかった場合、仮想サーバは bosco というデフォルト サーバ ファームに対して負荷が分散されます。

```
Router(config)# mod csm 2
Router(config-module-csm)# map map3 url
Router(config-slb-map-url)# match protocol http url *finance*
Router(config-slb-map-url)#
Router(config-slb-map-url)# map map4 url
Router(config-slb-map-url)# match protocol http url *mail*
Router(config-slb-map-url)#
Router(config-slb-map-url)# serverfarm bar1
Router(config-slb-sfarm)# real 10.1.0.105
Router(config-slb-real)# inservice
Router(config-slb-real)#
Router(config-slb-real)# serverfarm bar2
Router(config-slb-sfarm)# real 10.1.0.106
Router(config-slb-real)# inservice
Router(config-slb-real)#
Router(config-slb-real)# serverfarm bosco
Router(config-slb-sfarm)# real 10.1.0.107
Router(config-slb-real)# inservice
Router(config-slb-real)#
Router(config-slb-real)# policy pc1
Router(config-slb-policy)# serverfarm bar1
Router(config-slb-policy)# url-map map3
Router(config-slb-policy)# exit
Router(config-module-csm)#
Router(config-module-csm)# policy pc2
Router(config-slb-policy)# serverfarm bar2
Router(config-slb-policy)# url-map map4
Router(config-slb-policy)# exit
Router(config-module-csm)#
Router(config-module-csm)# vserver bar1
Router(config-slb-vserver)# virtual 10.1.0.86 tcp 80
Router(config-slb-vserver)# slb-policy pc1 priority 1
Router(config-slb-vserver)# slb-policy pc2 priority 4
Router(config-slb-vserver)# serverfarm bosco
Router(config-slb-vserver)# inservice
Router(config-slb-vserver)#
```

TCP パラメータの設定

Transmission Control Protocol（TCP）は、既知のプロトコル メッセージを使用して TCP セッションをアクティブおよび非アクティブにするコネクション型プロトコルです。サーバのロードバランシング中に接続データベースからの接続を追加または削除すると、最終ステート マシンによって SYN、SYN/ACK、FIN、RST などの TCP 信号が相互に関連付けられます。接続を追加すると、これらの信号を使用して、サーバの故障および回復の検出、サーバあたりの接続数の判別が行われます。

CSM は User Datagram Protocol（UDP）もサポートします。UDP はコネクション型でないため、通常は、（上位層プロトコルの詳細を取得せずに）プロトコル メッセージを見分けて、UDP メッセージ交換の開始または終了を検出することはできません。UDP 接続の終了は、設定可能なアイドル タイマーに基づいて検出されます。同じ実サーバに対して複数の同時接続を要求するプロトコル（FTP など）がサポートされています。仮想 IP アドレスを宛先とする Internet Control Management Protocol（ICMP）メッセージ（ping など）も処理されます。

TCP パラメータを設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# vserver virtserver-name`	仮想サーバを特定し、仮想サーバ コンフィギュレーション モードを開始します[1,2]。
ステップ 2	`Router(config-slb-vserver)# idle duration`	接続のパケット処理が行われていない場合に、接続情報を保持する時間（秒単位）を設定します[2]。*duration* の有効な値は 0（接続は無期限でオープンなまま維持される）～ 65535 秒で、デフォルトは 3600 秒です。 **（注）** **idle 0** を指定すると、接続は作成されますが、接続テーブルから自動的には削除されません。そのため、この接続を削除するまで、すべてのリソースが消費される可能性があります。最大接続数を指定するには、**INFINITE_IDLE_TIME_MAXCONNS** 環境変数を使用します。

1. モードまたはサブモードを終了するには、**exit** コマンドを入力します。メニューのトップ レベルである Router (config)> に戻るには、**end** コマンドを入力します。
2. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。

次に、仮想サーバ用の TCP パラメータを設定する例を示します。

```
Router(config-module-csm)# vserver barnett
Router(config-slb-vserver)# idle 10
```

CSM は、分割された TCP パケットをサポートします。TCP フラグメント機能は、レベル 4 ポリシーが定義されている VIP とだけ連動します。SYN パケットまたはレイヤ 7 ポリシーには作用しません。分割された TCP パケットをサポートするために、CSM は TCP フラグメントを既存のデータ フローに一致させるか、またはブリッジング VLAN ID に一致させます。CSM は、フラグメントをレイヤ 7 解析用に再構築しません。CSM には、バッファおよびフラグメント ID バケットの最終番号があるので、ハッシュ コリジョンが発生している場合はパケットを再送信する必要があります。

TCP スプライシングがイネーブルのときは、仮想サーバにレイヤ 7 ポリシーがない場合でも、仮想サーバをレイヤ 7 装置として指定する必要があります。このオプションは、TCP プロトコルにのみ有効です。

TCP スプライシングを設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# vserver virtserver-name`	仮想サーバを特定し、仮想サーバ コンフィギュレーション モードを開始します[1,2]。
ステップ 2	`Router(config-slb-vserver)# vserver tcp-protect`	TCP スプライシング用の仮想サーバを指定します[2]。
ステップ 3	`Router(config-slb-vserver)# virtual 100.100.100.100 tcp any service tcp-termination`	TCP スプライシングをイネーブルにします。

1. モードまたはサブモードを終了するには、**exit** コマンドを入力します。メニューのトップ レベルである Router (config)> に戻るには、**end** コマンドを入力します。
2. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。

部分的なサーバ ファーム フェールオーバーの設定

バックアップ サーバ ファームを設定すると、サーバ ファームを正常に維持するために必要なアクティブな実サーバ数、およびサーバ ファームを再度アクティブ化するために必要なアクティブ実サーバ数を指定する 2 つのスレッシュホールドを定義できます。

これらのスレッシュホールド値を指定しないと、サーバ ファーム内のすべての実サーバで障害発生時にサーバ ファーム全体が停止してしまいます。この場合、サーバ ファーム内の 1 台の実サーバが正常状態に戻ったときに、プライマリ サーバ ファームが再度動作可能になります。

部分的なサーバ ファーム フェールオーバーを設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# vserver` *virtserver-name*	仮想サーバを特定し、仮想サーバ コンフィギュレーション モードを開始します。
ステップ 2	`Router(config-slb-vserver)# serverfarm` *primary_serverfarm* [**backup** *backup_serverfarm* [**threshold outservice** *real_value* **inservice** *real_value*][**sticky**]]	デフォルトのサーバ ファームを仮想サーバに関連付け、バックアップ サーバ ファームを定義します。 **outservice** 値は、サーバ ファームを正常に維持するために最小限必要なアクティブ実サーバ数を指定します。**inservice** 値は、サーバ ファームを再度アクティブ化するために必要なアクティブ実サーバ数を指定します。 どちらの値も有効な範囲は、1 からサポートされる最大実サーバ数までです。**outservice** 値には、**inservice** 値より小さい値を指定する必要があります。

次に、サーバ ファーム内の正常な実サーバ数が 2 台以下になったときにバックアップ サーバ ファームがアクティブになり、サーバ ファーム内の正常な実サーバ数が 6 台になったときにプライマリ サーバ ファームが再度アクティブになるように設定する例を示します。

```
Router(config-slb-sfarm)# vserver barnett
Router(config-slb-vserver)# serverfarm bosco backup BACKUP threshold outservice 3
inservice 6
```

仮想サーバの依存関係の設定

仮想サーバを追跡するように CSM を設定できます。この機能を使うと、追跡対象の仮想サーバでサービスの停止または障害が発生すると、それに従属する仮想サーバも自動的に停止します。

仮想サーバの依存関係を設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# vserver` *dependent_virtserver_name*	従属する仮想サーバを特定し、仮想サーバ コンフィギュレーション モードを開始します。
ステップ 2	`Router(config-slb-vserver)# virtual` *ip-address* [*ip-mask*] *protocol port-number* [**service** {**ftp** | **rtsp** | **termination**}]	従属する仮想サーバの任意のポート番号または名前、接続カップリング、および接続タイプに対応する IP アドレスを設定します[2]。*protocol* 値は **tcp**、**udp**、**any**（ポート番号は不要）、または *number* 値（ポート番号は不要）です。
ステップ 3	`Router(config-slb-vserver)# status-tracking` *tracked_virtserver_name*	追跡対象の仮想サーバを特定します。この仮想サーバでサービスの停止または障害が発生すると、ステップ 1 で特定した従属する仮想サーバも自動的に停止します。

次に、仮想サーバ B でサービスの停止または障害が発生した場合に、仮想サーバ A および C が自動的に停止するように設定する例を示します。

```
Router(config-slb-sfarm)# vserver A
Router(config-slb-vserver)# virtual 10.1.0.85 tcp 80
Router(config-slb-vserver)# status-tracking B
Router(config-slb-vserver)# exit
Router(config-slb-sfarm)# vserver C
Router(config-slb-vserver)# virtual 10.1.0.86 tcp 80
Router(config-slb-vserver)# status-tracking B
```

リダイレクト仮想サーバの設定

redirect-vserver コマンドは、実サーバ専用の仮想サーバを設定するためのサーバ ファーム サブモード コマンドです。このマッピングを行うと、TCP セッションを介してクライアントが実サーバに永久的に接続されます。

リダイレクト仮想サーバを設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-slb-sfarm)#` **redirect-vserver** *name*	実サーバ専用の仮想サーバを設定し、リダイレクト サーバ サブモードを開始します[1,2]。
ステップ 2	`Router(config-slb-redirect-v)#` **webhost relocation** *relocation string*	このサーバ ファームに届いた HTTP 要求をリダイレクトする場合の、宛先 URL ホスト名を設定します。再割り当て文字列に指定できるのは、URL の先頭部分だけです。残りの部分は、元の HTTP 要求から取得されます[2]。
ステップ 3	`Router(config-redirect-v)#` **webhost backup** *backup string*	リダイレクト サーバのサービスが停止した場合に、HTTP 要求に応答して送信される再割り当て文字列を設定します。指定できるのは再割り当て文字列の先頭だけです。残りの部分は、元の HTTP 要求から取得されます[2]。
ステップ 4	`Router(config-redirect-v)#` **virtual** *v_ipaddress* **tcp** *port*	リダイレクト仮想サーバの IP アドレスおよびポートを設定します[2]。
ステップ 5	`Router(config-redirect-v)#` **idle** *duration*	リダイレクト仮想サーバの CSM 接続アイドル タイマーを設定します[2]。
ステップ 6	`Router(config-redirect-v)#` **client** *ip-address network-mask* [**exclude**]	リダイレクト仮想サーバにアクセスできるクライアントを制限するために使用する、IP アドレスおよびネットワーク マスクの組み合わせを設定します[2]。
ステップ 7	`Router(config-redirect-v)#` **inservice**	リダイレクト仮想サーバをイネーブルにし、アドバタイズを開始します[2]。
ステップ 8	`Router(config-redirect-v)#` **ssl** *port*	（任意）仮想サーバによる SSL 転送をイネーブルにします。
ステップ 9	`Router#` **show module csm vserver redirect** [**detail**]	設定されたリダイレクト サーバをすべて表示します。

1. モードまたはサブモードを終了するには、**exit** コマンドを入力します。メニューのトップ レベルに戻るには、**end** コマンドを入力します。
2. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。

次に、リダイレクト仮想サーバを設定して、サーバ ファーム内の実サーバに仮想サーバを指定する例を示します。

```
Router (config)# serverfarm FARM1
Router (config-slb-sfarm)# redirect-vserver REDIR_1
Router (config-slb-redirect-)# webhost relocation 127.1.2.30 301
Router (config-slb-redirect-)# virtual 172.1.2.30 tcp www
Router (config-slb-redirect-)# inservice
Router (config-slb-redirect-)# exit
Router (config-slb-sfarm)# redirect-vserver REDIR_2
Router (config-slb-redirect-)# webhost relocation 127.1.2.31 301
Router (config-slb-redirect-)# virtual 172.1.2.31 tcp www
Router (config-slb-redirect-)# inservice
Router (config-slb-redirect-)# exit
Router (config-slb-sfarm)# real 10.8.0.8
Router (config-slb-real)# redirect-vserver REDIR_1
Router (config-slb-real)# inservice
Router (config-slb-sfarm)# real 10.8.0.9
Router (config-slb-real)# redirect-vserver REDIR_2
Router (config-slb-real)# inservice
Router (config-slb-real)# end
Router# show module csm serverfarm detail
```

マップの設定

マップを作成して、複数の URL、Cookie、HTTP ヘッダー、および戻りコードをグループ内で定義すると、ポリシーを設定するときに、ポリシーをグループに関連付けることができます（「ポリシーの設定」[p.6-12] を参照）。URL の正規表現（*url1*、*url2* など）は、UNIX ファイル名の仕様に基づきます。詳細については、表 6-1 を参照してください。

URL マップを追加する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# map` *url-map-name* **url**	グループを作成し、複数の URL 一致条件を指定します[1,2]。
ステップ 2	`Router(config-slb-map-url)#` **match protocol http url** *url-path*	要求 URL と突き合せる文字列を指定します[2]。

1. モードまたはサブモードを終了するには、**exit** コマンドを入力します。メニューのトップ レベルに戻るには、**end** コマンドを入力します。
2. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。

表 6-1　文字列と一致する特殊文字

表記	説明
*	0 個以上の文字
?	1 文字
\	エスケープ文字
角かっこで囲まれた範囲 [0-9]	範囲内の任意の 1 文字と一致
範囲の先頭に ^ を付加	範囲内のどの文字とも一致しません。指定された文字以外のすべての文字と一致します。
.\a	アラート（ASCII 7）
.\b	バックスペース（ASCII 8）
.\f	フォーム フィード（ASCII 12）
.\n	改行（ASCII 10）
.\r	復帰（CR）（ASCII 13）
.\t	タブ（ASCII 9）
.\v	垂直タブ（ASCII 11）
.\0	ヌル（ASCII 0）
.\\	バックスラッシュ
.\x##	2 桁の 16 進表記で指定されたあらゆる ASCII 文字

Cookie マップを追加する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config)#` **map** *cookie-map-name* **cookie**	Cookie マップに複数の Cookie を設定します[1]。
ステップ 2	`Router(config-slb-map-cookie)#` **match protocol http cookie** *cookie-name* **cookie-value** *cookie-value-expression*	複数の Cookie を設定します。

1. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。

次に、マップを設定して、ポリシーを関連付ける例を示します。

```
Router(config-module-csm)# serverfarm pl_url_url_1
Router(config-slb-sfarm)# real 10.8.0.26
Router(config-slb-real)# inservice
Router(config-slb-real)# exit
Router(config-slb-sfarm)# exit
Router(config-slb-policy)# serverfarm pl_url_url_1
Router(config-slb-policy)# url-map url_1
Router(config-slb-policy)# exit
Router(config-module-csm)# serverfarm pl_url_url_2
Router(config-slb-sfarm)# real 10.8.0.27
Router(config-slb-real)# inservice
Router(config-slb-real)# exit
Router(config-slb-sfarm)# exit
Router(config-module-csm)# map url_1  url
Router(config-slb-map-url)# match protocol http url /url1
Router(config-slb-map-url)# exit
Router(config-module-csm)# map url_2 url
Router(config-slb-map-url)# match protocol http url /url/url/url
Router(config-slb-map-url)# match protocol http url /reg/*long.*
Router(config-slb-map-url)# exit
Router(config-module-csm)# policy policy_url_1
Router(config-module-csm)# policy policy_url_2
Router(config-slb-policy)# serverfarm pl_url_url_2
Router(config-slb-policy)# url-map url_2
Router(config-slb-policy)# exit
Router(config-module-csm)# vserver vs_url_url
Router(config-slb-vserver)# virtual 10.8.0.145 tcp 80
Router(config-slb-vserver)# slb-policy policy_url_1
Router(config-slb-vserver)# slb-policy policy_url_2
Router(config-slb-vserver)# inservice
Router(config-slb-vserver)# exit
```

map コマンドを使用して、HTTP ヘッダー タイプを指定し、マップ グループを作成します。**map** コマンドを入力すると、サブモードが開始され、要求で CSM に検索させるヘッダー フィールドおよび値を指定できます。

HTTP ヘッダー用のマップを作成する手順は、次のとおりです。

コマンド	目的
`Router(config-module-csm)# map name header`	HTTP ヘッダー マップ グループを作成して名前を指定します。

ヘッダー マップの詳細については、「一般ヘッダー解析の設定」（p.6-14）を参照してください。

戻りコードをチェックするためのマップを作成する手順は、次のとおりです。

コマンド	目的
`Router(config-module-csm)# map name retcode`	戻りコード マップ グループを作成して名前を指定します。

HTTP 戻りエラー コード チェックを設定する手順は、次のとおりです。

コマンド	目的
`Router(config-slb-sfarm)# retcode-map name_of_map`	HTTP 戻りエラー コード チェックを設定します。

戻りコード マップの詳細については、「HTTP 戻りコード チェックの設定」（p.9-11）を参照してください。

ポリシーの設定

ポリシーは、サーバ ファームにトラフィックを分散する場合に、トラフィックが満たさなければならないアクセス ルールです。ポリシーによって、CSM はレイヤ 7 トラフィックを分散させます。1 つの仮想サーバに複数のポリシーを割り当て、その仮想サーバに複数のアクセス ルールを作成できます。ポリシーを設定するときは、最初にアクセス ルール（マップ、クライアント グループ、および固定 [sticky] グループ）を設定してから、これらのアクセス ルールを所定のポリシーに従って結合します。

（注）　サーバ ファームとポリシーを関連付ける必要があります。サーバ ファームが関連付けられていないポリシーは、トラフィックを転送できません。ポリシーに関連付けられたサーバ ファームは、そのポリシーと一致するあらゆる要求を受信します。

ポリシーのマッチングを実行できる場合、CSM はポリシー リスト内の最初のポリシーを選択します。ポリシーは、仮想サーバにバインドされた順序でポリシー リストに配置されます。

関連付けられたサーバ ファーム内のすべてのサーバがダウンしている場合でも、ポリシーは一致されます。この場合のポリシーのデフォルト動作は、これらの接続を受け取らず、クライアントに reset（RST）を送り返すことです。この動作を変更する場合は、このポリシーにバックアップ サーバ ファームを追加します。

backup *backup_serverfarm* [**sticky**] オプションをバックアップ サーバ ファームに追加すると、プライマリ サーバ ファームに適用された固定（sticky）グループがバックアップ サーバ ファームにも適用されるかどうかも定義できます。プライマリ サーバ ファームのスティッキ性を指定しない場合、バックアップ サーバ ファームにその設定は適用されません。

たとえば、固定（sticky）グループにポリシーを設定した場合、このポリシー内のプライマリ サーバ ファームが固定されます。クライアントは、プライマリ サーバ ファームに設定された実サーバに固定されます。プライマリ サーバ ファームのすべての実サーバで障害が発生した場合、このクライアントからの新規要求はバックアップ サーバ ファームに送信されます。プライマリ サーバ ファームの実サーバが復旧して運用可能になれば、次のアクションが結果として実行されます。

- バックアップ実サーバへの既存の接続は、バックアップ実サーバによって継続されます。
- 固定（sticky）オプションがバックアップ サーバ ファームでイネーブルの場合、クライアントからの新規要求がバックアップ実サーバに送信されます。
- 固定（sticky）オプションがバックアップ サーバ ファームで使用されていなければ、新規要求がプライマリ実サーバに送信されます。

リスト内のポリシーを並べ替えるには、ポリシーを削除してから、正しい順番で再入力します。ポリシーを削除して入力するには、仮想サーバ サブモードで **no slb-policy** *policy name* コマンドおよび **slb-policy** *policy name* コマンドを入力します。

ロードバランシング ポリシーを設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# policy policy-name`	ポリシーを作成し、ポリシー サブモードを開始して、ポリシー属性を設定します[1]。
ステップ 2	`Router(config-slb-policy)# url-map url-map-name`	URL マップをポリシーに関連付けます[2]。**map** コマンドを使用して、事前に URL マップ /Cookie マップを作成および設定しておく必要があります。「一般ヘッダー解析の設定」（p.6-14）を参照してください。
ステップ 3	`Router(config-slb-policy)# cookie-map cookie-map-name`	Cookie マップをポリシーに関連付けます[2]。
ステップ 4	`Router(config-slb-policy)# header-map name`	HTTP ヘッダー マップをポリシーに関連付けます。
ステップ 5	`Router(config-slb-policy)# sticky-group group-id`	このポリシーを特定の固定（sticky）グループに関連付けます[2]。
ステップ 6	`Router(config-slb-policy)# client-group value | std-access-list-name`	ポリシーに対応するクライアント フィルタを設定します。クライアント フィルタを定義する場合に使用するのは、標準 IP アクセス リストだけです。
ステップ 7	`Router(config-slb-policy)# serverfarm primary_serverfarm [backup backup_serverfarm [threshold {inservice real_value} [outservice real_value]][sticky]]`	特定のロードバランシング ポリシーを処理するプライマリ サーバ ファームを設定します。1 つのポリシーに設定できるサーバ ファームは 1 つだけです[2]。サーバ ファームのスレッシュホールドの設定の詳細については、「部分的なサーバ ファーム フェールオーバーの設定」（p.6-7）を参照してください。
ステップ 8	`Router(config-slb-policy)# set ip dscp dscp-value`	パケットがロードバランシング ポリシーと一致した場合は、トラフィックに Differentiated Services Code Point（DSCP）値をマークします[2]。
ステップ 9	`Router(config-slb-policy)# nat client {client-pool-name | static}`	（任意）NAT モード クライアントをイネーブルにします[2]（「クライアント NAT プールの設定」[p.5-8]を参照）。 **（注）** サーバファームとポリシーの両方にクライアント NAT が設定されている場合は、ポリシーがサーバファームより優先されます。

1. モードまたはサブモードを終了するには、**exit** コマンドを入力します。メニューのトップ レベルに戻るには、**end** コマンドを入力します。
2. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。

次の例は、map1 という URL マップがすでに設定されていることが前提です。サーバ ロードバランシング ポリシーを設定し、仮想サーバに関連付ける例を示します。

```
Router(config-slb-policy)# serverfarm pl_sticky
Router(config-slb-sfarm)# real 10.1.0.105
Router(config-slb-sfarm)# inservice
Router(config-slb-policy)# exit
Router(config-module-csm)# policy policy_sticky_ck
Router(config-slb-policy)# serverfarm pl_sticky
Router(config-slb-policy)# url-map map1
Router(config-slb-policy)# exit
Router(config-module-csm)# vserver vs_sticky_ck
Router(config-slb-vserver)# virtual 10.1.0.80 tcp 80
Router(config-slb-vserver)# slb-policy policy_sticky_ck
Router(config-slb-sfarm)# inservice
Router(config-slb-policy)# exit
```

一般ヘッダー解析の設定

ソフトウェア Release 2.1(1) から、CSM は一般 HTTP 要求ヘッダー解析をサポートしています。HTTP 要求ヘッダーには、ユーザの要件に合わせてコンテンツをどのようにフォーマットするかを記述するフィールドがあります。

一般ヘッダー解析の概要

CSM は HTTP ヘッダーのフィールドを解析し、照合することによって得た情報をポリシー情報とともに使用して、ロードバランシングの決定を下します。たとえば、HTTP ヘッダーのブラウザ タイプ フィールドを解析することによって、CSM はユーザがモバイル ブラウザでコンテンツにアクセスしているかどうかを判別し、モバイル ブラウザ用にフォーマットされたコンテンツのあるサーバを選択できます。

HTTP get 要求ヘッダー レコードの 1 例を示します。

```
GET /?u HTTP/1.1<0D><0A>
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg<0D><0A>
Referer: http://www.yahoo.com/<0D><0A>
Accept-Language: en-us<0D><0A>
Accept-Encoding: gzip, deflate<0D><0A>
User-Agent: Mozilla/4.0 (compatible; MSIE 5.0; Windows NT; DigExt)<0D><0A>
Host: finance.yahoo.com<0D><0A>
Connection: Keep-Alive<0D><0A>
Cookie: B=51g3cjstaq3vm; Y=1<0D><0A>
<0D><0A>
```

一般ヘッダー解析の設定例

一般ヘッダー解析を設定するには、HTTP ヘッダーのフィールドに対するポリシー マッチングの実行を CSM に指示するコマンドを入力します。次に、CSM 上で一般ヘッダー解析を設定する方法について説明します。

- HTTP ヘッダー用マップの作成（p.6-15）
- ヘッダー フィールドおよび一致する値の指定（p.6-15）
- ポリシーへの HTTP ヘッダー マップの割り当て（p.6-15）
- 仮想サーバへのポリシーの割り当て（p.6-16）
- 一般ヘッダー解析の設定例（p.6-16）

HTTP ヘッダー用マップの作成

map コマンドを使用して、HTTP ヘッダー タイプを指定し、マップ グループを作成します。**map** コマンドを入力すると、サブモードが開始され、要求で CSM に検索させるヘッダー フィールドおよび値を指定できます。

HTTP ヘッダー用のマップを作成する手順は、次のとおりです。

コマンド	目的
`Router(config-module-csm)# map` *name* `header`	HTTP ヘッダー マップ グループを作成して名前を指定します。

(注)　その他のマップ タイプには、URL および Cookie があります。

HTTP のヘッダー挿入は、CSM にクライアント IP アドレスの HTTP ヘッダーへの挿入のような情報を挿入させる機能です。HTTP ヘッダーの挿入はヘッダー マップ内で設定します。設定の詳細については、「HTTP ヘッダー挿入」（p.8-20）を参照してください。

ヘッダー フィールドおよび一致する値の指定

match コマンドを使用して、フィールド名を指定し、さらに HTTP 要求を受信したときに CSM が照合する値を指定します。

ヘッダー フィールドおよび一致する値を指定する手順は、次のとおりです。

コマンド	目的
`Router(config-slb-map-header)# match protocol http header` *field* `header-value` *expression*	フィールド名および値を指定します。フィールドは Cookie 以外の任意の HTTP ヘッダーにできます。Cookie ヘッダーを設定しなければならない場合は、Cookie マップを設定できます。

(注)　CSM では、ポリシー マッチングの条件として、1 つまたは複数の HTTP ヘッダー フィールドを指定できます。1 つの HTTP ヘッダー グループで複数のフィールドを設定した場合、この条件を満たすにはグループのすべての式が一致しなければなりません。

ポリシーへの HTTP ヘッダー マップの割り当て

ポリシー サブモードで、そのポリシーに含めるヘッダー マップを指定します。ヘッダー マップには、ポリシーに含める HTTP ヘッダー条件を指定します。

ポリシーに HTTP ヘッダー マップを割り当てる手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# policy` *policy-name*	ポリシーを作成します。
ステップ 2	`Router(config-slb-policy)# header-map` *name*	HTTP ヘッダー マップをポリシーに割り当てます。

（注）　デフォルトでは、あらゆる HTTP ヘッダー情報がポリシー規則を満たします。HTTP URL および HTTP Cookie は、それぞれ固有のヘッダー情報タイプであり、CSM によって別個に処理されます。

仮想サーバへのポリシーの割り当て

仮想サーバ サブモードで、**vserver** *virtserver-name* コマンドを使用し、ヘッダー マップが割り当てられているポリシーの名前を指定します。

ヘッダー マップが割り当てられたポリシー名を指定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# vserver virtserver-`*name*	仮想サーバを設定します。
ステップ 2	`Router(config-slb-policy)# header-map` *name*	HTTPヘッダー マップをポリシーに割り当てます。

一般ヘッダー解析の設定例

次に、一般ヘッダー解析を設定する例を示します。

```
Router(config)# mod csm 2
Router(config-module-csm)# !!!configure generic header map
Router(config-module-csm)# map map2 header
Router(config-slb-map-heaer)# $col http header Host header-value *.yahoo.com

Router(config-slb-map-header)# !!! configure serverfarm
Router(config-slb-map-header)# serverfarm farm2
Router(config-slb-sfarm)# real 10.1.0.105
Router(config-slb-real)# inservice
Router(config-slb-real)# exit
Router(config-slb-sfarm)# exit

Router(config-module-csm)# !!! configurate policy
Router(config-module-csm)# policy pc2
Router(config-slb-policy)# serverfarm farm2
Router(config-slb-policy)# header-map map2
Router(config-slb-policy)# exit

Router(config-module-csm)# !!! config vserver
Router(config-module-csm)# vserver vs2
Router(config-slb-vserver)# virtual 10.1.0.82 tcp 80
Router(config-slb-vserver)# slb-policy pc2
Router(config-slb-vserver)# inservice
Router(config-slb-vserver)# end
Router(config)# show module csm 2 map det
```

冗長性の設定

この章では、冗長性の設定方法について説明します。

フォールトトレランスの設定

ここでは、フォールトトレラントの設定について説明します。この設定では、2 つの異なる Catalyst 6500 シリーズ シャーシのそれぞれに Content Switching Module（CSM; コンテント スイッチング モジュール）を搭載します。

（注） 1 つの Catalyst 6500 シリーズ シャーシに CSM を 2 つ搭載して、フォールトトレラント コンフィギュレーションを作成することもできます。フォールトトレラントは、セキュア（ルータ）モードまたは非セキュア（ブリッジ）モードのどちらでも設定できます。

セキュア（ルータ）モードでは、クライアント側およびサーバ側 VLAN（仮想 LAN）によって、CSM とクライアント側のルータとの間、および CSM とサーバ側のサーバとの間にフォールトトレラントな（冗長性のある）接続パスが確立されます。冗長構成では、2 つの CSM はアクティブ CSM およびスタンバイ CSM として機能します。各 CSM には同じ IP、仮想サーバ、サーバ プール、および実サーバ情報が格納されます。各 CSM は、クライアント側およびサーバ側ネットワークで同じように設定されます。ネットワークは、フォールトトレラント コンフィギュレーションを単一の CSM として認識します。

（注） 複数のフォールトトレラント CSM ペアを設定する場合、同じフォールトトレラント VLAN を使用する CSM ペアを複数設定しないでください。フォールトトレラント CSM ペアごとに異なるフォールトトレラント VLAN を使用してください。

フォールトトレランスを設定するための条件は、次のとおりです。

- Catalyst 6500 シリーズ シャーシに 2 つの CSM が搭載されている。
- 2 つの CSM が同じ設定である。一方の CSM はアクティブとして、他方はスタンバイとして設定します。
- 各 CSM が同じクライアント側およびサーバ側 VLAN に接続されている。
- CSM 間の通信が共有プライベート VLAN によって提供されている。
- ネットワークが冗長 CSM を単一のエンティティとして認識している。
- 1 GB/ 秒の容量のリンク設定によって接続の冗長性が得られる。CSM のステート変化に正しい時刻のスタンプが与えられるように、スイッチの Cisco IOS ソフトウェアでカレンダーをイネーブルにします。

  カレンダーをイネーブルにするには、次のコマンドを使用します。

```
Cat6k-2# configure terminal
Cat6k-2(config)# clock timezone WORD offset from UTC
Cat6k-2(config)# clock calendar-valid
```

各 CSM のクライアント側およびサーバ側 VLAN には異なる IP アドレスが設定されているので、CSM はネットワークにヘルス モニタ プローブを送信して（「ヘルス モニタリング用プローブの設定」[p.9-2] を参照）、応答を受信できます。アクティブおよびスタンバイの両方の CSM が、動作中にプローブを送信します。パッシブ CSM が制御している場合、その CSM は受信したプローブ応答からサーバのステータスを認識します。

接続複製機能は、非 TCP 接続と TCP 接続の両方をサポートします。CSM に複製を設定するには、仮想サーバ モードで **replicate csrp**{**sticky** | **connection**} コマンドを入力します。

（注）　**replicate** コマンドは、ディセーブルがデフォルトの設定です。

接続の冗長性を得るために接続複製機能を使用する場合には、次のコマンドを使用します。

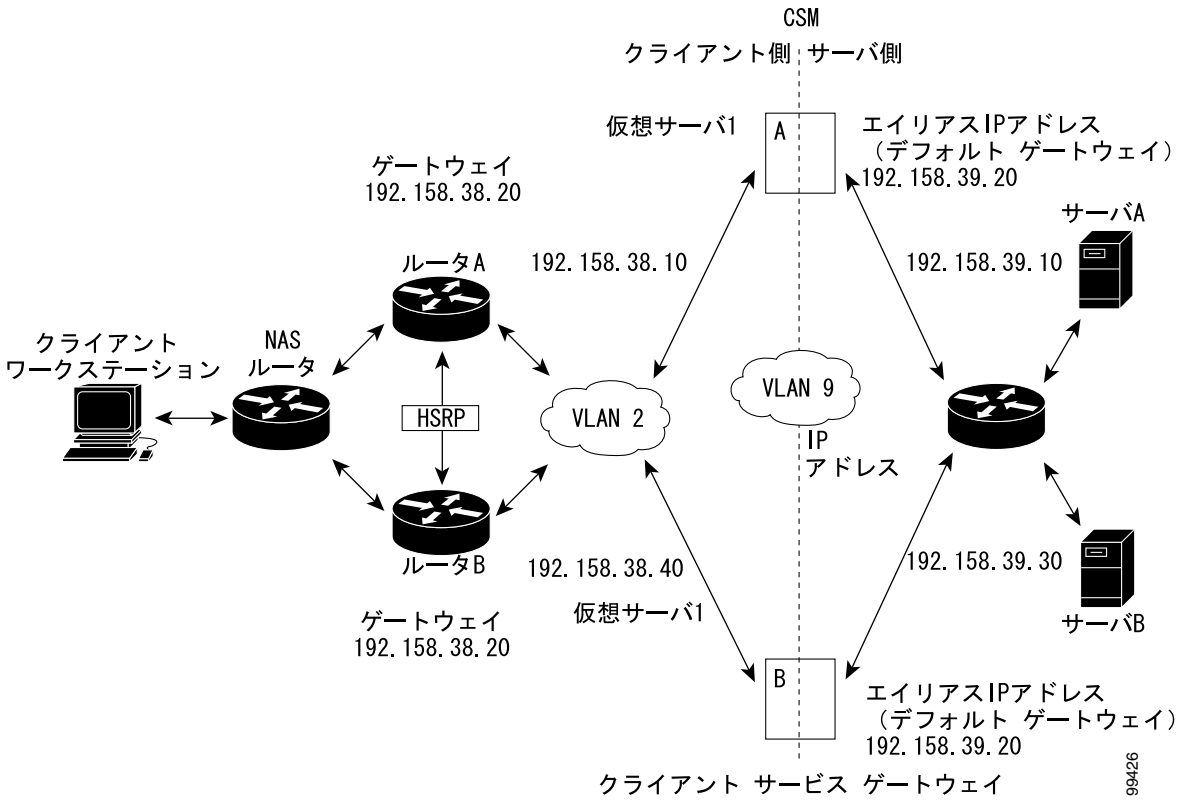
```
Cat6k-2# configure terminal
Cat6k-2(config)# no ip igmp snooping
```

複製フレームには、ユニキャスト IP アドレスと同様マルチキャスト タイプの宛先 MAC があるので、**no ip igmp snooping** コマンドを入力する必要があります。スイッチが、Internet Group Management Protocol（IGMP）にマルチキャスト グループ メンバーシップの検索およびマルチキャスト Forwarding Information Base（FIB; 転送情報ベース）の確立をリスンする場合、スイッチはグループ メンバーを検索せずにマルチキャスト テーブルをプルーニングします。アクティブからスタンバイまですべてのマルチキャスト フレームは、エラー結果の原因となるため廃棄されます。

サーバ側 VLAN にルータがない場合、各サーバのデフォルト ルートはエイリアス IP アドレスを示します。

図 7-1 に、セキュア（ルータ）モードのフォールトトレラント コンフィギュレーションの設定方法を示します。

図 7-1　フォールトトレラントの設定



（注）　図 7-1 のアドレスは、次の 2 種類の手順に関連します。

アクティブ（A）CSM をフォールトトレランスとして設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# vlan 2 client`	クライアント側 VLAN 2 を作成し、SLB VLAN モードを開始します[1]。
ステップ 2	`Router(config-slb-vlan-client)# ip addr 192.158.38.10 255.255.255.0`	VLAN 2 にコンテント スイッチング IP アドレスを割り当てます。
ステップ 3	`Router(config-slb-vlan-client)# gateway 192.158.38.20`	（任意）クライアント側 VLAN のゲートウェイを HSRP 対応ゲートウェイとして定義します。
ステップ 4	`Router(config-module-csm)# vserver vip1`	仮想サーバを作成し、SLB vserver モードを開始します。
ステップ 5	`Router(config-slb-vserver)# virtual 192.158.38.30 tcp www`	仮想 IP アドレスを作成します。
ステップ 6	`Router(config-module-csm)# inservice`	サーバをイネーブルにします。
ステップ 7	`Router(config-module-csm)# vlan 3 server`	サーバ側 VLAN 3 を作成し、SLB VLAN モードを開始します。
ステップ 8	`Router(config-slb-vlan-server)# ip addr 192.158.39.10 255.255.255.0`	VLAN 3 に CSM の IP アドレスを割り当てます。
ステップ 9	`Router(config-slb-vlan-server)# alias ip addr 192.158.39.20 255.255.255.0`	VLAN 3 のデフォルト ルートを割り当てます。
ステップ 10	`Router(config-slb-vlan-server) vlan 9`	VLAN 9 をフォールトトレラント VLAN として定義します。
ステップ 11	`Router(config-module-csm)# ft group` *`ft-group-number`* `vlan 9`	コンテント スイッチングのアクティブおよびスタンバイ（A/B）グループである VLAN 9 を作成します。
ステップ 12	`Router(config-module-csm)# vlan`	VLAN モードを開始します[1]。
ステップ 13	`Router(vlan)# vlan 2`	クライアント側 VLAN 2 を設定します[2]。
ステップ 14	`Router(vlan)# vlan 3`	サーバ側 VLAN 3 を設定します。
ステップ 15	`Router(vlan)# vlan 9`	フォールトトレラントの VLAN 9 を設定します。
ステップ 16	`Router(vlan)# exit`	**exit** コマンドを入力して、設定を有効にします。

1. モードまたはサブモードを終了するには、**exit** コマンドを入力します。メニューのトップ レベルに戻るには、**end** コマンドを入力します。
2. デフォルトの設定に戻すには、このコマンドの no 形式を使用します。

スタンバイ（B）CSM をフォールトトレランスとして設定する手順は、次のとおりです（図 7-1 を参照）。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# vlan 2 client`	クライアント側 VLAN 2 を作成し、SLB VLAN モードを開始します[1]。
ステップ 2	`Router(config-slb-vlan-client)# ip addr 192.158.38.40 255.255.255.0`	VLAN 2 にコンテント スイッチング IP アドレスを割り当てます。
ステップ 3	`Router(config-slb-vlan-client)# gateway 192.158.38.20`	クライアント側 VLAN ゲートウェイを定義します。
ステップ 4	`Router(config-module-csm)# vserver vip1`	仮想サーバを作成し、SLB 仮想サーバ モードを開始します。
ステップ 5	`Router(config-slb-vserver)# virtual 192.158.38.30 tcp www`	仮想 IP アドレスを作成します。
ステップ 6	`Router(config-module-csm)# inservice`	サーバをイネーブルにします。
ステップ 7	`Router(config-module-csm)# vlan 3 server`	サーバ側 VLAN 3 を作成し、SLB VLAN モードを開始します。
ステップ 8	`Router(config-slb-vserver)# ip addr 192.158.39.30 255.255.255.0`	VLAN 3 に CSM の IP アドレスを割り当てます。
ステップ 9	`Router(config-slb-vserver)# alias 192.158.39.20 255.255.255.0`	VLAN 2 のデフォルト ルートを割り当てます。
ステップ 10	`Router(config-module-csm) vlan 9`	VLAN 9 をフォールトトレラント VLAN として定義します。
ステップ 11	`Router(config-module-csm)# ft group` *ft-group-number* **vlan 9**	CSM アクティブおよびスタンバイ（A/B）グループである VLAN 9 を作成します。
ステップ 12	`Router(config-module-csm)# show module csm all`	フォールトトレラント システムのステートを表示します。

1. モードまたはサブモードを終了するには、**exit** コマンドを入力します。メニューのトップ レベルに戻るには、**end** コマンドを入力します。

HSRP の設定

ここでは Hot Standby Router Protocol（HSRP）の設定を概説し（図 7-2 を参照）、Catalyst 6500 シリーズ スイッチで HSRP および CSM フェールオーバーを指定して CSM を設定する方法について説明します。

HSRP の設定の概要

図 7-2 では、HSRP ゲートウェイ（10.100.0.1）を介してクライアント側ネットワーク（10.100/16）から内部 CSM クライアント ネットワーク（10.6/16、VLAN 136）にルーティングするように、Catalyst 6500 シリーズ スイッチのスイッチ 1 およびスイッチ 2 を設定しています。設定には次の点に留意してください。

- クライアント側ネットワークには、HSRP ID 2 という HSRP グループ ID が割り当てられています。
- 内部 CSM クライアント ネットワークには、HSRP ID 1 という HSRP グループ ID が割り当てられています。

（注） HSRP グループ 1 は、HSRP グループ 2 のクライアント ネットワーク ポートを追跡できるように、トラッキングをオンにしておく必要があります。HSRP グループがこれらのポートのアクティブ ステート変化を検出すると、その変化を複製し、HSRP アクティブ スイッチ（スイッチ 1）と HSRP スタンバイ スイッチ（スイッチ 2）の両方で、同じネットワーク情報が共有されるようにします。

この設定例では、クライアント側とサーバ側 VLAN の間でトラフィックを転送するように 2 つの CSM（スイッチ 1 およびスイッチ 2 に 1 つずつ）が設定されています。

- クライアント VLAN 136

  （注） クライアント VLAN は、実際は内部 CSM VLAN ネットワークです。実際のクライアント ネットワークはスイッチの反対側にあります。

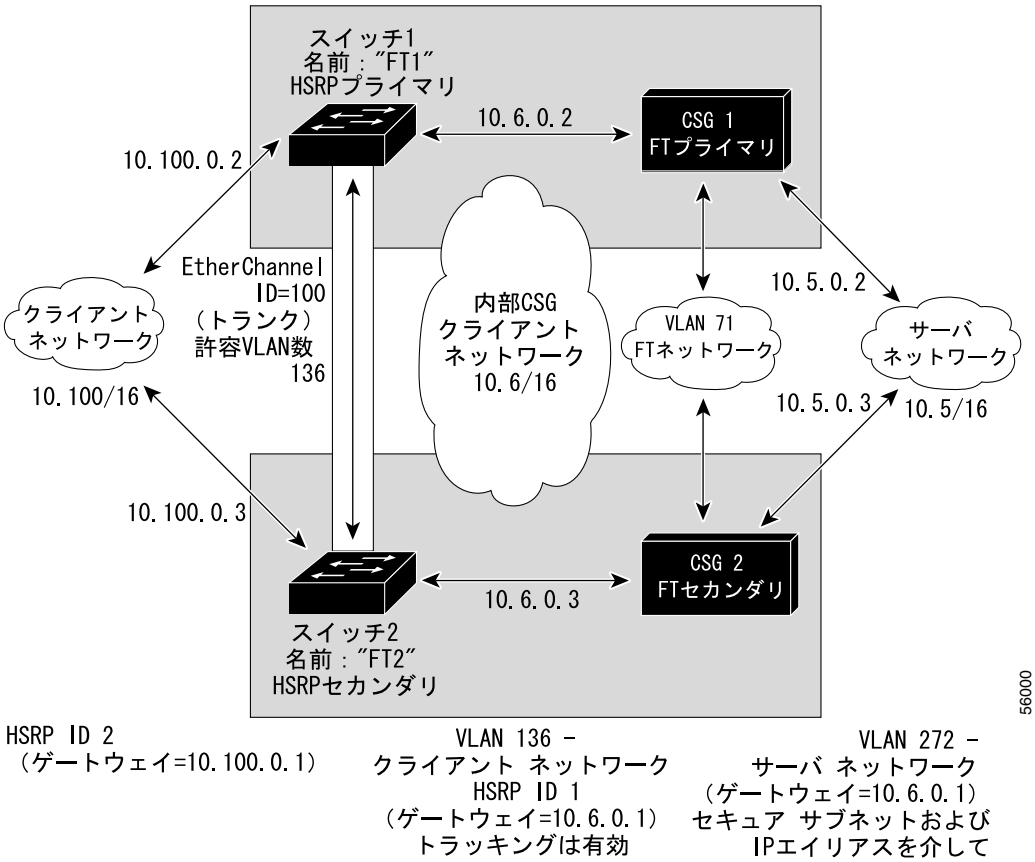
- サーバ VLAN 272

  サーバ ネットワーク（10.5/1）上の実際のサーバは、エイリアス ゲートウェイ（10.5.0.1）を介して CSM サーバ ネットワークを示しているため、サーバは安全なサブネットを実現できます。

  設定例では、EtherChannel はトランキングがイネーブルになるようにセットアップされているため、内部 CSM クライアント ネットワークのトラフィックを 2 つの Catalyst 6500 シリーズ スイッチ間で送受信できます。図 7-2 に設定を示します。

  （注） EtherChannel はアクティブ スイッチへのリンクの切断、および CSM 以外のスイッチ コンポーネントの故障に対して保護します。また、スイッチ内のアクティブな CSM と別のスイッチとの間でパスを確立して、CSM とスイッチが独立してフェールオーバーできるようにします。これによって、フォールトトレランス レベルがさらに向上します。

図 7-2　HSRP の設定



HSRP ゲートウェイの作成

ここでは、クライアント側ネットワーク用の HSRP ゲートウェイを作成する方法について説明します。クライアント側ネットワークのゲートウェイは HSRP ID 2 です。

（注） この例では、HSRP はファスト イーサネット ポート 3/6 に設定されています。

HSRP ゲートウェイを作成する手順は、次のとおりです。

ステップ 1　スイッチ 1 ― FT1（HSRP アクティブ）を設定します。

```
Router(config)# interface FastEthernet3/6
Router(config)# ip address 10.100.0.2 255.255.0.0
Router(config)# standby 2 priority 110 preempt
Router(config)# standby 2 ip 10.100.0.1
```

ステップ 2　スイッチ 2 ― FT2（HSRP スタンバイ）を設定します。

```
Router(config)# interface FastEthernet3/6
Router(config)# ip address 10.100.0.3 255.255.0.0
Router(config)# standby 2 priority 100 preempt
Router(config)# standby 2 ip 10.100.0.1
```

フォールトトレラント HSRP コンフィギュレーションの作成

ここでは、フォールトトレラント HSRP セキュア モードを設定する方法について説明します。非セキュア モードを設定するには、次の例外に従ってコマンドを入力します。

- サーバ側およびクライアント側 VLAN の両方に同じ IP アドレスを割り当てます。
- サーバ側 VLAN にデフォルト ゲートウェイを割り当てる場合は、**alias** コマンドを使用しないでください。

フォールトトレラント HSRP コンフィギュレーションを作成する手順は、次のとおりです。

ステップ 1　HSRP FT1 の VLAN を設定します。

```
Router(config)# module csm 5
Router(config-module-csm)# vlan 136 client
Router(config-slb-vlan-client)# ip address 10.6.0.245 255.255.0.0
Router(config-slb-vlan-client)# gateway 10.6.0.1
Router(config-slb-vlan-client)# exit

Router(config-module-csm)# vlan 272 server
Router(config-slb-vlan-server)# ip address 10.5.0.2 255.255.0.0
Router(config-slb-vlan-server)# alias 10.5.0.1 255.255.0.0
Router(config-slb-vlan-server)# exit

Router(config-module-csm)# vlan 71

Router(config-module-csm)# ft group 88 vlan 71
Router(config-slb-ft)# priority 30
Router(config-slb-ft)# preempt
Router(config-slb-ft)# exit

Router(config-module-csm)# interface Vlan136
ip address 10.6.0.2 255.255.0.0
standby 1 priority 100 preempt
standby 1 ip 10.6.0.1
standby 1 track Fa3/6 10
```

ステップ 2　HSRP FT2 の VLAN を設定します。

```
Router(config)# module csm 6
Router(config-module-csm)# vlan 136 client
Router(config-slb-vlan-client)# ip address 10.6.0.246 255.255.0.0
Router(config-slb-vlan-client)# gateway 10.6.0.1
Router(config-slb-vlan-client)# exit

Router(config-module-csm)# vlan 272 server
Router(config-slb-vlan-server)# ip address 10.5.0.3 255.255.0.0
Router(config-slb-vlan-server)# alias 10.5.0.1 255.255.0.0
Router(config-slb-vlan-server)# exit

Router(config-module-csm)# vlan 71

Router(config-module-csm)# ft group 88 vlan 71
Router(config-slb-ft)# priority 20
Router(config-slb-ft)# preempt
Router(config-slb-ft)# exit

Router(config-module-csm)# interface Vlan136
ip address 10.6.0.3 255.255.0.0
standby 1 priority 100 preempt
standby 1 ip 10.6.0.1
standby 1 track Fa3/6 10
```

（注）　トラッキングを作動させるには、preempt をオンにする必要があります。

ステップ 3　両方のスイッチで EtherChannel を設定します。

```
Router(console)# interface Port-channel100
Router(console)# switchport
Router(console)# switchport trunk encapsulation dot1q
Router(console)# switchport trunk allowed vlan 136
```

（注）　デフォルトでは、ポート チャネル上ですべての VLAN が許可されます。

ステップ 4　問題が発生しないように、サーバおよびフォールトトレラントの CSM VLAN を削除します。

```
Router(console)# switchport trunk remove vlan 71
Router(console)# switchport trunk remove vlan 272
```

ステップ 5　EtherChannel にポートを追加します。

```
Router(console)# interface FastEthernet3/25
Router(console)# switchport
Router(console)# channel-group 100 mode on
```

インターフェイスおよびデバイスのトラッキングの設定

フォールトトレラント HSRP を設定した場合、CSM のアクティブおよびスタンバイ状態は、アクティブ HSRP グループの状態とは一致しません。アクティブ HSRP があるシャーシ内にあり、アクティブ CSM が別のシャーシ内にある場合、トラフィックは 2 つのシャーシ間のトランク ポートを経由します。

トラッキングを設定して、HSRP グループ、物理インターフェイス、およびゲートウェイの状態を追跡できます。

HSRP グループのトラッキング

HSRP グループのトラッキングは、指定した追跡対象グループの HSRP 状態が変化したときに、Cisco IOS ソフトウェアが CSM にアクティブ スイッチオーバーの実行を指示するメッセージを送信するように設定できます。

HSRP グループのトラッキングを設定する手順は、次のとおりです。フォールトトレラント サブモードで実行します。

コマンド	目的
`Router(config-slb-ft)# track group group_number`	追跡対象の HSRP グループを指定します。

ゲートウェイのトラッキング

ゲートウェイのトラッキングを設定すると、Cisco IOS ソフトウェアが設定済みのゲートウェイ IP アドレスとネクスト ホップ IP アドレスを CSM へ送信します。その後、CSM はゲートウェイのアベイラビリティを定期的に調べます。ゲートウェイが使用不可能な場合、CSM は強制的にアクティブ スイッチオーバーを実行します。

ゲートウェイのトラッキングを設定する手順は、次のとおりです。フォールトトレラント サブモードで実行します。

コマンド	目的
`Router(config-slb-ft)# track gateway ip_addr`	追跡対象のゲートウェイIPアドレスを指定します。

インターフェイスのトラッキング

インターフェイスのトラッキングは、指定した物理インターフェイスがダウンしたときに、Cisco IOS ソフトウェアが CSM にアクティブ スイッチオーバーの実行を指示するメッセージを送信するように設定できます。

インターフェイスのトラッキングを設定する手順は、次のとおりです。フォールトトレラント サブモードで実行します。

コマンド	目的
`Router(config-slb-ft)# track interface {async | ctunnel | dialer | fastethernet | gigabitethernet}`	追跡対象のインターフェイスを指定します。

トラッキング モードの設定

トラッキング モードを設定する手順は、次のとおりです。フォールトトレラント サブモードで実行します。

コマンド	目的
`Router(config-slb-ft)# track mode {any | all}`	トラッキング モードを指定します。 **any** キーワードは、追跡中のデバイスのいずれかがダウンするか、HSRP 状態がスタンバイに変わった場合に、強制的にスイッチオーバーを実行します。 **all** キーワードは、設定済みのすべてのトラッキング機能（グループ、ゲートウェイ、およびインターフェイス）に関して追跡中のデバイスが 1 つでもダウンすると、強制的にスイッチオーバーを実行します。

接続の冗長性の設定

接続の冗長性によって、アクティブ CSM で障害が発生し、スタンバイ CSM がアクティブになるときに、オープンな接続が応答を停止する事態を防止できます。接続の冗長性によって、アクティブ CSM からスタンバイ CSM へのフェールオーバー時にオープンなままにしておくべき各接続について、アクティブ CSM はスタンバイ CSM に転送情報をコピーします。

接続の冗長性を設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router# configure terminal`	ルータ コンフィギュレーション モードを開始します。
ステップ 2	`Router(config)# no ip igmp snooping`	設定から IGMP スヌーピングを削除します。
ステップ 3	`Router(config-module-csm)# vserver virtserver-name`	仮想サーバを特定し、仮想サーバ サブモードを開始します。
ステップ 4	`Router(config-slb-vserver)# virtual ip-address [ip-mask] protocol port-number [service ftp]`	仮想サーバの属性を設定します。
ステップ 5	`Router(config-slb-vserver)# serverfarm serverfarm-name`	サーバ ファームと仮想サーバを関連付けます。
ステップ 6	`Router(config-slb-vserver)# sticky duration [group group-id] [netmask ip-netmask]`	同じクライアントからの接続には同じ実サーバが使用されるようにします。
ステップ 7	`Router(config-slb-vserver)# replicate csrp sticky`	スティッキの複製をイネーブルにします。
ステップ 8	`Router(config-slb-vserver)# replicate csrp connection`	接続の複製をイネーブルにします。
ステップ 9	`Router(config-slb-vserver)# inservice`	仮想サーバのロードバランシングをイネーブルにします。
ステップ 10	`Router(config-module-csm)# ft group group-id vlan vlanid`	フォールトトレランスを設定し、フォールトトレランス サブモードを開始します。
ステップ 11	`Router(config-slb-ft)# priority value`	CSM のプライオリティを設定します。
ステップ 12	`Router(config-slb-ft)# failover failover-time`	スタンバイ CSM がアクティブ CSM になるまでの待機時間を設定します。
ステップ 13	`Router(config-slb-ft)# preempt`	オンラインになったときに、プライオリティの高いCSMがフォールトトレラント グループを制御するようにします。

次に、接続の冗長性を目的として、フォールトトレランスを設定する例を示します。

```
Router(config-module-csm)# vserver VS_LINUX-TELNET
Router(config-slb-vserver)# virtual 10.6.0.100 tcp telnet
Router(config-slb-vserver)# serverfarm SF_NONAT
Router(config-slb-vserver)# sticky 100 group 35
Router(config-slb-vserver)# replicate csrp sticky
Router(config-slb-vserver)# replicate csrp connection
Router(config-slb-vserver)# inservice
Router(config-slb-vserver)# exit
Router(config-module-csm)# ft group 90 vlan 111
Router(config-slb-ft)# priority 10
Router(config-slb-ft)# failover 3
Router(config-slb-ft)# preempt
Router(config-slb-ft)# exit
```

設定の同期化

1 つのシャーシ内または別々のシャーシ内のアクティブ CSM とスタンバイ CSM 間で設定を同期化できます。同期化はフォールトトレラント VLAN 上で実行されます。

（注） フォールトトレラント VLAN 上のトラフィックはブロードキャスト パケットを使用します。そのため、アクティブ CSM とスタンバイ CSM 間の通信に必要なデバイス以外は、すべてのデバイスをフォールトトレラント VLAN から削除することを推奨します。

（注） ここに示す手順を記述されているとおりに実行することが重要です。設定を同期化する前に（下記のステップ 4 で説明されているとおり）アクティブ CSM で **alt** *standby_ip_address* コマンドを入力しなかった場合、バックアップ CSM の VLAN IP アドレスは削除されます。

アクティブ CSM 上で同期化を設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router# configure terminal`	ルータ コンフィギュレーション モードを開始します。
ステップ 2	`Router(config)# module csm` *slot-number*	アクティブ CSM のスロット番号を指定します。
ステップ 3	`Router(config-module-csm)# vlan` *vlan_ID* `{client | server}`	クライアント側およびサーバ側 VLAN を設定します。
ステップ 4	`Router(config-slb-vlan-client)# ip addr` *active_ip_addr netmask* **alt** *standby_ip_addr netmask*	この特定の VLAN 上の CSM に IP アドレスを設定します。**alt** キーワードを入力して、スタンバイ CSM へ送信される代替 IP アドレスを指定します。 **（注）** 設定を同期化する前にアクティブ CSM で **alt** *standby_ip_address* コマンドを入力しなかった場合、バックアップ CSM の VLAN IP アドレスは削除されます。
ステップ 5	`Router(config-slb-vlan-client)# exit`	VLAN コンフィギュレーション モードを終了します。
ステップ 6	`Router(config-module-csm)# ft group` *group-id* **vlan** *vlanid*	フォールトトレランスを設定し、フォールトトレランス サブモードを開始します。
ステップ 7	`Router(config-slb-ft)# priority` *active_value* **alt** *standby_value*	CSM のプライオリティを設定します。**alt** キーワードを入力して、スタンバイ CSM へ送信される代替プライオリティの値を指定します。

スタンバイ CSM 上で同期化を設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router# configure terminal`	ルータ コンフィギュレーション モードを開始します。
ステップ 2	`Router(config)# module csm` *slot-number*	スタンバイ CSM のスロット番号を指定します。
ステップ 3	`Router(config-module-csm)# ft group` *group-id* `vlan` *vlanid*	フォールトトレランスを設定し、フォールトトレラント VLAN を指定します。

同期化を設定する手順は、次のとおりです。これらの手順はアクティブ CSM 上で実行します。

コマンド	目的
`Router# hw-module csm` *slot-number* `standby config-sync`	設定を同期化します。このコマンドは、設定を同期化するたびに入力します。

次に、両方の CSM に同期化を設定する例を示します。

- アクティブ CSM：

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# module csm 5
Router(config-module-csm)# vlan 130 client
Router(config-slb-vlan-client)# ip addr 123.44.50.5 255.255.255.0 alt 123.44.50.7
255.255.255.0
Router(config-slb-vlan-client)# gateway 123.44.50.1
Router(config-slb-vlan-client)# exit
Router(config-module-csm)# vlan 150 server
Router(config-slb-vlan-server)# ip addr 123.46.50.6 255.255.255.0 alt 123.44.40.8
255.255.255.0
Router(config-slb-vlan-server)# alias 123.60.7.6 255.255.255.0
Router(config-slb-vlan-server)# route 123.50.0.0 255.255.0.0 gateway 123.44.50.1
Router(config-slb-vlan-server)# exit
Router(config-module-csm)# ft group 90 vlan 111
Router(config-slb-ft)# priority 10 alt 15
Router(config-slb-ft)# end
```

- スタンバイ CSM：

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# module csm 6
Router(config-module-csm)# ft group 90 vlan 111
Router(config-slb-ft)# end
```

次に、アクティブ CSM とスタンバイ CSM 間で設定を同期化する例を示します。

```
Router# hw-module csm 5 standby config-sync
%CSM_SLB-6-REDUNDANCY_INFO:Module 5 FT info:Active:Bulk sync started
%CSM_SLB-6-REDUNDANCY_INFO:Module 5 FT info:Active:Manual bulk sync completed
```

ヒットレス アップグレードの設定

*ヒットレス アップグレード*機能を使用すると、アップグレードのためのダウンタイムが原因で主要なサービスが停止する事態を招かずに、新しいバージョンにアップグレードできます。ヒットレス アップグレードを設定する手順は、次のとおりです。

ステップ 1　preempt がイネーブルになっている場合は、オフにします。

ステップ 2　スタンバイ CSM で write memory を実行します。

ステップ 3　新しいリリースでスタンバイ CSM をアップグレードし、CSM を再起動します。

スタンバイ CSM は、新しいリリースのスタンバイとして起動します。スティッキ バックアップをイネーブルにしている場合、スタンバイ CSM は 5 分間以上、スタンバイ モードが続きます。

ステップ 4　アクティブ CSM をアップグレードします。

ステップ 5　アクティブ CSM を再起動します。

アクティブ CSM を再起動すると、スタンバイ CSM が新しいアクティブ CSM になり、サービスを引き受けます。

ステップ 6　再起動した CSM はスタンバイ CSM として起動します。

CHAPTER 8

追加機能およびオプションの設定

この章では、コンテント スイッチングの設定方法について説明します。

セッションの持続性（スティッキ性）の設定

セッションの固定（スティッキ性）は、同一のクライアントから同一のサーバへ常時複数の（同時または連続した）接続を送信する機能です。この機能は、特定のロードバランシング環境で一般的に必要です。

アプリケーションのトランザクションを完了する（ブラウザで Web サイトにアクセスし、購入するさまざまな品目を選択してからチェックアウトするなど）には、通常、複数（ときには何百、何千）の同時または連続接続が必要です。こういったトランザクションの多くは、一時的に重要な情報を生成して、それを使用します。この情報はトランザクションを処理する特定のサーバ上で保存したり修正したりします。このトランザクションの完了までに数分から数時間かかる可能性があり、クライアントはその間同じサーバに何度も送信する必要があります。

バックエンド共有データベースによる多層設計でも問題の一部は解決できますが、ローカル サーバ キャッシュを利用することで、アプリケーションのパフォーマンスをさらに改善できます。ローカル サーバのキャッシュを使用するとデータベースに接続する必要がなくなり、新しいサーバが選択されるたびにトランザクション固有の情報が得られます。

持続性に関する問題の中でも最も難しいのは、複数の接続にわたって個々のクライアントをどのように特定するかということです。ロードバランシング デバイスはクライアントの識別に使用できるあらゆる情報を保存し、現在トランザクションを処理しているサーバにその情報を関連付ける必要があります。

（注） Content Switching Module（CSM; コンテント スイッチング モジュール）は、256,000 のエントリからなるスティッキ データベースを維持できます。

CSM は個々のクライアントを識別して、次の方法で固定処理を実行します。

- 送信元 IP アドレスの固定

  CSM に送信元 IP アドレス全体（32 ビットのネットマスクを含めて）を学習させるか、またはその一部を学習させるかを設定できます。

- SSL 識別情報の固定

  クライアントおよびサーバが Secure Socket Layer（SSL）を介して通信している場合、複数の接続にわたって一意の SSL 識別番号が維持されます。SSL バージョン 3.0 または Transport Layer Security（TLS）1.0 では、クリア テキストでこの識別番号を伝送する必要があります。CSM はこの値を使用することによって、個々のトランザクションを識別できます。ただし、この SSL ID は再度ネゴシエートできるため、SSL ID を常に正しいサーバに固定できるわけではありません。SSL ID ベースの固定方式を利用すると、常に SSL ID を再利用させることによって、SSL 終端装置のパフォーマンスが向上します。

  （注） CSM を Catalyst 6500 SSL モジュールと組み合わせて使用した場合、各 Catalyst 6500 SSL モジュールの MAC アドレスが特定のオフセットで SSL ID 内に挿入されるため、SSL ID の再ネゴシエート後も SSL ID を固定できます。この固定方法は、仮想サーバのコンフィギュレーション サブモードで **ssl-sticky** コマンドを使用して設定できます。

  スティッキ接続の設定情報については、『*Catalyst 6500 Series SSL Services Module Configuration Note*』の Chapter 5「Configuring Different Modes of Operation」を参照してください。

  ssl-sticky コマンドについては、『*Catalyst 6500 Series Switch Content Switching Module Command Reference*』を参照してください。

- ダイナミック Cookie ラーニング

  特定の Cookie 名を探して、クライアント要求の HTTP ヘッダーまたはサーバの「Set-Cookie」メッセージから自動的にその値を学習するように CSM を設定できます。

  CSM はデフォルトで、Cookie 値全体を学習します。この機能は CSM ソフトウェア Release 4.1.(1) において、オプションのオフセットおよび長さを取り入れて拡張され、Cookie 値の一部分だけを学習するように CSM に対して指示できるようになりました。「Cookie 固定のオフセットおよび長さ」（p.8-4）を参照してください。

  ダイナミック Cookie ラーニングは、同一の Cookie 内にセッション ID またはユーザ ID を複数保存するアプリケーションを扱う場合に役立ちます。スティッキ性に関連があるのは、Cookie 値の特定のバイトだけです。

  CSM ソフトウェア Release 4.1(1) には、ダイナミック Cookie スティッキ機能も追加されています。これは、URL の一部としての Cookie 情報を検索する（さらに学習して固定する）機能です。詳細については、「URL ラーニング」（p.8-5）を参照してください。URL の学習は、HTTP URL に Cookie 情報を組み込むアプリケーションで有用です。場合によっては、この機能を使用して Cookie を拒否するクライアントに対処できます。

- Cookie 挿入

  CSM はサーバに代わって Cookie を挿入するので、サーバが Cookie を設定しない場合でも Cookie 固定を実行できます。Cookie には、CSM が特定の実サーバ固定を確実に実行するための情報が含まれています。

- HTTP ヘッダーの固定

  CSM は、Mobile Station ISDN Number（MSISDN; モバイル ステーション ISDN 番号）、サービス キー、セッション ID などの HTTP ヘッダー情報の内容に基づいて固定を実行します。

固定（sticky）グループの設定

固定（sticky）グループを設定するには、固定方法（送信元 IP、SSL ID、Cookie、または HTTP ヘッダー）とそのグループのパラメータを設定し、さらにポリシーと関連付ける必要があります。固定（sticky）タイムアウトは、スティッキ情報がスティッキ テーブルで維持される期間を指定します。デフォルトの固定タイムアウト値は 1440 分（24 時間）です。特定のエントリの固定（sticky）タイマーは、そのエントリに一致する新規接続が開かれるたびにリセットされます。

（注） 複数のポリシーまたは仮想サーバは、潜在的に同じ固定グループに設定できます。その場合、それらのポリシーまたは仮想サーバへのすべての接続が固定処理されます。ポリシー 1 および 2、または仮想サーバ 1 および 2 が同じ固定グループに設定されている場合、ポリシー 1 または仮想サーバ 1 を介してサーバ A に固定されるクライアントが、ポリシー 2 または仮想サーバ 2 を介して同一のサーバ A に固定されるため、これらの接続は「buddy 接続」とも呼ばれます。

注意 複数のポリシーまたは仮想サーバで同じ固定グループを使用している場合、すべてのポリシーまたはサーバが間違いなく、同じサーバ ファームか、またはグループ内で同じサーバを指定している異なるサーバ ファームを使用していることが重要です。

固定グループを設定する手順は、次のとおりです。

コマンド	目的
Router(config-module-csm)# **sticky** *sticky-group-id* {**netmask** *netmask* | **cookie** *name* | **ssl** | **header** *name*} [**address** [**source** | **destination** | **both**]][**timeout** *sticky-time*]	同じポリシーと一致する同じクライアントからの接続で、同じ実サーバが使用されるようにします[1]。

1. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。

次に、固定グループを設定して、ポリシーに関連付ける例を示します。

```
Router(config-module-csm)# sticky 1 cookie foo timeout 100
Router(config-module-csm)# serverfarm pl_stick
Router(config-slb-sfarm)# real 10.8.0.18
Router(config-slb-real)# inservice
Router(config-slb-sfarm)# real 10.8.0.19
Router(config-slb-real)# inservice
Router(config-slb-real)# exit
Router(config-slb-sfarm)# exit
Router(config-module-csm)# policy policy_sticky_ck
Router(config-slb-policy)# serverfarm pl_stick
Router(config-slb-policy)# sticky-group 1
Router(config-slb-policy)# exit
Router(config-module-csm)# vserver vs_sticky_ck
Router(config-slb-vserver)# virtual 10.8.0.125 tcp 90
Router(config-slb-vserver)# slb-policy policy_sticky_ck
Router(config-slb-vserver)# inservice
Router(config-slb-vserver)# exit
```

Cookie 挿入

サーバが現在適切な Cookie を設定していない場合にセッション Cookie を固定するには、Cookie 挿入機能を使用します。この機能をイネーブルにすると、CSM はクライアントからサーバへの応答に Cookie を挿入します。CSM は次にサーバからクライアントへのトラフィック フローに Cookie を挿入します。

次に、固定するために Cookie を指定する例を示します。

```
Router(config-module-csm)# sticky 5 cookie mycookie insert
```

Cookie 固定のオフセットおよび長さ

Cookie 値は、クライアント / サーバ間のトランザクションで、一部を残して変更される場合があります。その場合、特定サーバへの連続接続を保つために、変更されない部分を使用できます。接続の連続性を固定または維持するために、Cookie の変更されない部分を **cookie offset** *num* [**length** *num*] コマンドを使用して、オフセットおよび長さの値で指定できます。

オフセット（Cookie 値を先頭のバイトから数える）および長さ（Cookie で使用する部分の長さ）をバイトで指定し、固定接続の維持に使用します。これらの値はスティッキ テーブルに保存されます。

オフセットおよび長さは 0 ～ 4000 バイトの範囲で指定できます。Cookie 値がオフセットより長く、オフセットと Cookie の長さを足したものより短い場合、CSM はオフセットの後ろの Cookie 部分に応じて接続を固定します。

次に、Cookie のオフセットおよび長さを設定する例を示します。

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# module csm 4
Router(config-module-csm)# sticky 20 cookie SESSION_ID
Router(config-slb-sticky-cookie)# cookie offset 10 length 6
```

URL ラーニング

URL ラーニングによる Cookie 固定機能により、CSM は Set-Cookie フィールドまたは URL 埋め込み Cookie のセッション情報をキャプチャできます。CSM は、サーバ応答の Set-Cookie HTTP ヘッダーに埋め込まれた特定の Cookie 値に基づいて、固定テーブルのエントリを作成します。

URL ラーニングを設定すると、CSM は次の 3 通りの方法で Cookie 値を学習できます。

- サーバからクライアント方向に設定された Cookie メッセージ
- クライアント要求内の Cookie
- URL に埋め込まれた Cookie 値

最初の 2 つの方法は標準のダイナミック Cookie ラーニング機能ですでにサポートされています。3 番めの方法は URL ラーニング機能として追加されました。

多くの場合、そのあとの一連の HTTP 要求内において、クライアントは同じ Cookie 値を戻します。CSM は、それに一致する値に基づいて同じサーバにクライアントを固定します。ただし、クライアントによってはブラウザで Cookie をディセーブルにしているため、このタイプの Cookie 固定接続ができない場合もあります。URL Cookie ラーニングの新機能で、CSM は URL ストリングに埋め込まれた Cookie 名および値を抽出できます。この機能が実行されるのは、サーバが Web ページの URL リンクに Cookie を埋め込んでいる場合だけです。

クライアント要求に Cookie が含まれていない場合、CSM は、CSM に設定されたセッション ID ストリング（?session-id=）を探します。このストリングに対応する値が、CSM がキャッシュ内で探しているセッション ID 番号です。セッション ID は、要求情報が保存されていて、なおかつクライアント要求の送信先であるサーバと一致します。

セッション Cookie および URL セッション ID は異なる可能性があるため、Cisco IOS の **sticky** *id* **cookie** *name* コマンドがアップデートされました。次の例で、正しい構文を示します。

（注）　このリリースの Cookie 固定オフセット機能をサポートするために、アップデートされたコマンドにはオフセットおよび長さを指定する構文が含まれています。「Cookie 固定のオフセットおよび長さ」（p.8-4）を参照してください。

クライアント / サーバの動作およびそのフレーム シーケンスに応じて、HTTP Cookie、Set-Cookie ヘッダー、または URL 埋め込み Cookie で表示されている同一の Cookie 値が、標準の HTTP Cookie に表示される場合があります。また、Cookie が URL に埋め込まれているか、HTTP Cookie ヘッダーに表示されているかによって、Cookie 名は URL によって異なる場合があります。異なる名前の Cookie および URL は、これらの 2 つのパラメータの多くがサーバ上で別々に設定されるために生じます。次に、Set-Cookie 名の例を示します。

```
Set-Cookie: session_cookie = 123
```

次に、URL の例を示します。

```
http://www.example.com/?session-id=123
```

sticky コマンドの *name* フィールドには、Cookie ヘッダーに表示される Cookie 名を指定します。このコマンドに追加された **secondary** *session_id* には、URL に表示される対応する Cookie 名を指定します。

次に、URL ラーニング機能の設定例を示します。

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# module csm 4
Router(config-module-csm)# sticky 30 cookie session_cookie
Router(config-slb-sticky-cookie)# cookie secondary session-id
Router(config-slb-sticky-cookie)#
```

HTTP ヘッダーの固定

クライアントがプロキシ デバイスの後ろに配置されている場合、送信元 IP アドレスなどの一部の情報が失われます。HTTP ヘッダーの内容（たとえば、MSISDN、サービス キー、セッション ID など）に基づいてクライアント セッションを一意に識別できます。指定した HTTP ヘッダーの内容に基づいて固定を実行するように CSM を設定できます。

HTTP ヘッダーの固定を設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router (config-module-csm)# sticky sticky-group-id header header_name timeout timeout`	HTTP ヘッダーの固定をイネーブルにし、スティッキヘッダー サブモードを開始します。
ステップ 2	`Router (config-slb-sticky-header)# header offset offset length length`	（任意）*offset* オプションは、ヘッダーの先頭から無視するバイト数を指定します。*length* オプションは、ヘッダー内で解析するバイト数を指定します。

次に、HTTP ヘッダーの固定を設定し、ヘッダー オフセットと長さを指定する例を示します。

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# module csm 4
Router(config-module-csm)# sticky 10 header msisdn timeout 20
Router(config-slb-sticky-header)# header offset 5 length 50
```

RHI の設定

ここでは Route Health Injection（RHI）の設定方法について説明します。

RHI について

ここでは、RHI について説明します。

RHI の概要

RHI は、CSM に VIP アドレスの可用性をネットワーク全体にアドバタイズさせます。また、ネットワーク全体にわたって同一の VIP アドレスおよびサービスを持つ複数の CSM 装置を配置できます。ある CSM は、ほかの装置でサーバ ロードバランス サービスを利用できなくなった場合に、ほかの装置のサービスを変更できます。この CSM は、ほかのサーバ ロードバランシング デバイスよりクライアント システムに論理上近いので、サービスの提供もできます。

（注） CSM は VIP アドレスをホスト ルートとしてアドバタイズしますが、ほとんどのルータはホスト ルート情報をインターネットに伝播しないので、RHI の用途はイントラネットに限定されます。

RHI をイネーブルにするには、次のように CSM を設定します。

- 実サーバをプローブし、使用可能な仮想サーバおよび VIP アドレスを識別します。
- 変更が発生するたびに、VIP アドレスの可用性情報を Multilayer Switch Feature Card（MSFC）に正確にアドバタイズします。

（注） 電源投入時に RHI がイネーブルの場合、各 VIP アドレスが使用可能になるので、CSM は MSFC にメッセージを送信します。

MSFC は RHI が提供する VIP アドレスの可用性情報を定期的に伝播します。

（注） セキュリティ上の理由から、ほとんどのルータはホスト　ルート情報をインターネットに伝播しないので、通常、RHI の用途はイントラネットに限定されます。

RHI を使用しない VIP アドレスへのルーティング

RHI を使用しない場合、トラフィックは VIP アドレスが属すクライアント VLAN へのルートを経由して、VIP アドレスに送信されます。CSM の電源を投入すると、MSFC はルーティング テーブルにクライアント VLAN へのルートを作成し、このルート情報をほかのルータと共有します。VIP に到達するために、クライアント システムはルータを使用して、各 VIP アドレスが属すネットワーク サブネット アドレスに要求を送信します。

サブネットまたはセグメントに到達可能であっても、その場所にある CSM の仮想サーバが動作していない場合、要求は失敗します。ほかの CSM 装置はさまざまな場所に配置することができます。ただし、ルータは単に論理的な距離に基づいてサブネットに要求を送信します。

RHI を使用しないと、VIP アドレスを使用できるかどうかが検証されずに、トラフィックが VIP アドレスに送信されます。この場合は、VIP に接続された実サーバがアクティブではないこともあります。

（注） デフォルトでは、CSM は設定された VIP アドレスをアドバタイズしません。

RHI を使用する VIP アドレスへのルーティング

RHI を使用すると、VIP アドレスが使用可能になって、使用不可能な VIP アドレスのアドバタイズが取り消された場合に、CSM は MSFC にアドバタイズを送信します。ルータはルーティング テーブルを参照して、要求をクライアントから VIP アドレスに送信するために必要なパス情報を検索します。RHI 機能が有効な場合、一致した中で最も固有性の高いものが VIP アドレス情報としてアドバタイズされます。クライアントに対する要求は、アクティブな VIP サービスを使用して、CSM に到達するパスを経由して送信されます。

VIP アドレスのインスタンスが複数存在する場合、クライアント ルータは VIP アドレスのインスタンスごとに必要な情報（可用性およびホップ数）を受信して、その VIP アドレスに対する最適なルートを判別できます。ルータは CSM が論理上、クライアント システムに近くなるようにパスを選択します。

（注） CSM はコンテンツを処理するすべての実サーバをプローブで精査することによって、指定された VIP アドレスに到達できるかどうかを判別します。したがって、RHI を使用する場合はプローブも設定する必要があります。VIP アドレスに到達できるかどうかを判別したあと、CSM はこの可用性情報を MSFC と共有します。次に、MSFC はこの VIP 可用性情報をイントラネットのほかの装置に伝播します。

CSM が VIP の可用性を判別する仕組み

VIP が使用可能かどうかを CSM が判断できるようにするには、プローブ（HTTP、ICMP、Telnet、TCP、FTP、SMTP、または DNS）を設定し、それをサーバ ファームに関連付ける必要があります。プローブが設定されている場合、CSM は次の確認を行います。

- プローブ用に設定されたすべてのサーバ　ファーム上のすべての実サーバをプローブで調べます。
- 到達可能な（到達可能な実サーバを 1 台以上含む）サーバ ファームを識別します。
- 到達可能な（到達可能なサーバ ファームを 1 つ以上含む）仮想サーバを識別します。
- 到達可能な（到達可能な仮想サーバを 1 台以上含む）VIP を識別します。

VIP の可用性情報の伝播

RHI を使用する場合、CSM は使用可能な VIP アドレスを含むアドバタイズ メッセージを MSFC に送信します。MSFC は、CSM から受信する VIP アドレスごとに、ルーティング テーブルにエントリを追加します。MSFC で動作中のルーティング プロトコルは、ほかのルータにルーティング テーブル アップデートを送信します。VIP アドレスが使用不可能になると、そのルートはアドバタイズされなくなり、エントリはタイムアウトし、ルーティング プロトコルは変更を伝播します。

（注） RHI を CSM で動作させるには、CSM が搭載されているシャーシ内の MSFC で Cisco IOS Release 12.1.7(E) 以降を稼働させ、その MSFC をクライアント側ルータとして設定する必要があります。

仮想サーバ用 RHI の設定

仮想サーバ用の RHI を設定する手順は、次のとおりです。

ステップ 1　VLAN が設定されていることを確認します（第 4 章「VLAN の設定」を参照）。

ステップ 2　プローブをサーバ ファームに関連付けます（「ヘルス モニタリング用プローブの設定」[p.9-2] を参照）。

ステップ 3　実サーバをプローブで調べるように CSM を設定します（「ヘルス モニタリング用プローブの設定」[p.9-2] を参照）。

ステップ 4　**advertise active** SLB 仮想サーバ コマンドを入力して、仮想サーバごとに RHI をイネーブルにします。

```
Router(config-module-csm)# vserver virtual_server_name
Router(config-slb-vserver)# advertise active
```

次に、vserver1 という名前の仮想サーバに対して RHI をイネーブルにする例を示します。

```
Router(config-module-csm)# vserver vserver1
Router(config-slb-vserver)# advertise active
```

環境変数

variable *name string* コマンドを使用して、コンフィギュレーションの環境変数をイネーブルにできます。表 8-1 で CSM 環境変数の値について説明します。

表 8-1　CSM 環境変数

名前	デフォルト	有効値	説明
ARP_INTERVAL	300	整数 （15 ～ 31536000）	設定したホストの ARP 要求の間隔（秒）
ARP_LEARNED_INTERVAL	14400	整数 （60 ～ 31536000）	学習したホストの ARP 要求の間隔（秒）
ARP_GRATUITOUS_INTERVAL	15	整数 （10 ～ 31536000）	gratuitous ARP 要求の間隔（秒）
ARP_RATE	10	整数（1 ～ 60）	ARP 再試行の間隔（秒）
ARP_RETRIES	3	整数（2 ～ 15）	ホスト ダウンのフラグを立てる前に ARP を再試行する回数
ARP_LEARN_MODE	1	整数（0 ～ 1）	CSM が応答のみ（0）の MAC アドレスを学習するか、すべてのトラフィック（1）の MAC アドレスを学習するかを指定します。
ARP_REPLY_FOR_NO_INSERVICE_VIP	0	整数（0 ～ 1）	CSM が停止している実サーバの ARP に応答（1）するかどうかを指定します。
ADVERTISE_RHI_FREQ	10	整数（1 ～ 65535）	CSM が RHI アップデートをチェックする頻度（秒）
AGGREGATE_BACKUP_SF_STATE_TO_VS	0	整数（0 ～ 1）	仮想サーバの状態に、バックアップ サーバ ファームの動作可能状態を含めるかどうかを指定します。
COOKIE_INSERT_EXPIRATION_DATE	Fri, 1 Jan 2010 01:01:50 GMT	ストリング （2 ～ 63 文字）	CSM によって挿入される HTTP Cookie の期限切れの時刻および日付を設定します。
CSM_FAST_FIN_TIMEOUT	10	整数（10 ～ 65535）	FIN が検出された後、接続をリセットするためのタイムアウト（秒）を指定します。
DEST_UNREACHABLE_MASK	65535	整数（0 ～ 65535）	ICMP 宛先到達不能コードの転送をビットマスクで定義します。
FT_FLOW_REFRESH_INT	60	整数（1 ～ 65535）	フォールトトレラントのスロー パス フローのリフレッシュ間隔（秒）
HTTP_CASE_SENSITIVE_MATCHING	1	整数（0 ～ 1）	URL（Cookie、ヘッダー）の一致および固定で、大文字と小文字を区別するかどうかを指定します。
HTTP_URL_COOKIE_DELIMITERS	/?&#+	ストリング （1 ～ 64 文字）	URL ストリングの Cookie の区切り文字のリストを設定します。
INFINITE_IDLE_TIME_MAXCONNS	1024	0 ～ 4294967295	無限アイドル時間を指定できる最大接続数を設定します。
MAX_PARSE_LEN_MULTIPLIER	1	整数（1 ～ 16）	設定した max-parse-len をこの総数で乗算します。

表 8-1　CSM 環境変数（続き）

名前	デフォルト	有効値	説明
MAX_VSERVERS_PER_VIP	10	整数（7 ～ 10）	同じIPアドレスを保持できる仮想サーバの最大数を指定します。値は 2 の乗数（たとえば、2^7=128、2^10=1024）として指定されます。
MSTS_RDP_VIP_LIST	なし	ストリング（最大 256 バイト）	Microsoft Terminal Services Remote Desktop Protocol（MSTS-RDP）をサポートする VIP のリストを設定します。
NAT_CLIENT_HASH_SOURCE_PORT	0	整数（0 ～ 1）	送信元ポートを使用してクライアントの NAT IP アドレスを取得するかどうかを指定します。
NO_RESET_UNIDIRECTIONAL_FLOWS	0	整数（0 ～ 1）	設定されている場合、タイムアウト時に単一方向フローをリセットしないように指定します。
REAL_SLOW_START_ENABLE	3	整数（0 ～ 10）	スロースタート サーバへ送信される平均接続数を指定し、スロースタート機能をディセーブルまたはイネーブルにします。値は 2 の乗数として指定されます。
ROUTE_UNKNOWN_FLOW_PKTS	0	整数（0 ～ 2）	既存のフローと一致しない SYN または non-SYN パケットをルーティングするかどうかを指定します。
SECURE_HTTP_PRIV_KEY_FILE	なし	ストリング（0 ～ 256 文字）	HTTP サーバが使用するプライベート キー ファイルを指定します。
SECURE_HTTP_PORT	443	整数（1 ～ 65535）	HTTPS サーバのポート番号を指定します。
SECURE_HTTP_SERVER_CERTIFICATE	なし	ストリング（0 ～ 256 文字）	HTTPS サーバが使用する証明書ファイルを指定します。
SECURE_HTTP_SSL_METHOD	0	整数（0 ～ 3）	HTTPSサーバが使用するSSLのバージョンを指定します。
SECURE_HTTP_TFTP_HOST_IPADDRESS	なし	ストリング（0 ～ 16 文字）	HTTP サーバの証明書を持つ TFTP サーバの IP アドレスを指定します。 この変数が設定されていない場合は、MSFC がデフォルト TFTP サーバになり、CSM は暗黙で MSFC ファイルシステム上の証明書を検索します。
SECURE_SASP_ENABLE	0	整数（0 ～ 1）	セキュア Server Application State Protocol（SASP）機能をイネーブル（1）にするかどうかを指定します。
SECURE_SASP_SSL_METHOD	0	整数（0 ～ 3）	SASP クライアントが使用する SSL のバージョンを指定します。
SECURE_SASP_TFTP_HOST_IPADDRESS	なし	ストリング（0 ～ 16 文字）	SASP クライアントの証明書を持つ TFTP サーバの IP アドレスを指定します。この変数が設定されていない場合は、MSFC がデフォルトの TFTP サーバになり、CSM は暗黙的に MSFC ファイルシステム上の証明書を検索します。

表 8-1　CSM 環境変数（続き）

名前	デフォルト	有効値	説明
SECURE_SASP_SERVER_CERTIFICATE	なし	ストリング（0 ～ 256 文字）	セキュア SASP クライアントが使用する証明書ファイルを指定します。
SECURE_SASP_PRIV_KEY_FILE	なし	ストリング（0 ～ 256 文字）	セキュア SASP クライアントが使用するプライベート キー ファイルを指定します。
SWITCHOVER_RP_ACTION	0	整数（0 ～ 1）	スーパバイザ エンジン Route Processor（RP）のスイッチオーバーが発生したあとで、復旧（0）または停止 / 再起動（1）するかどうかを指定します。
SWITCHOVER_SP_ACTION	0	整数（0 ～ 1）	スーパバイザ エンジン Switch Processor（SP）のスイッチオーバーが発生したあとで、復旧（0）または停止 / 再起動（1）するかどうかを指定します。
SYN_COOKIE_INTERVAL	3	整数（1 ～ 60）	新しい Syn-Cookie キーが生成される間隔を指定します（秒）。
SYN_COOKIE_THRESHOLD	5000	整数（0 ～ 1048576）	Syn-Cookie の動作のスレッシュホールドを指定します（中断されているセッション数）。
TCP_MSS_OPTION	1460	整数（1 ～ 65535）	レイヤ 7 の処理に対して CSM が送信できる最大セグメント サイズ（MSS）値を指定します。
TCP_WND_SIZE_OPTION	8192	整数（1 ～ 65535）	レイヤ 7 の処理に対して CSM が送信できるウィンドウ サイズ値を指定します。
VSERVER_ICMP_ALWAYS_RESPOND	false	ストリング（1 ～ 5 文字）	「true」の場合、仮想サーバの状態に関わらず ICMP プローブに応答します。
XML_CONFIG_AUTH_TYPE	Basic	ストリング（5 ～ 6 文字）	Xml-Config に対して HTTP 認証タイプを指定します。Basic または Digest です。

コンフィギュレーションの環境変数を表示する例を示します。

```
Router# show mod csm 5 variable

variable                          value
----------------------------------------------------------------
ARP_INTERVAL                      300
ARP_LEARNED_INTERVAL              14400
ARP_GRATUITOUS_INTERVAL           15
ARP_RATE                          10
ARP_RETRIES                       3
ARP_LEARN_MODE                    1
ARP_REPLY_FOR_NO_INSERVICE_VIP    0
ADVERTISE_RHI_FREQ                10
AGGREGATE_BACKUP_SF_STATE_TO_VS   0
COOKIE_INSERT_EXPIRATION_DATE     Fri, 1 Jan 2010 01:01:50 GMT
DEST_UNREACHABLE_MASK             0xffff
FT_FLOW_REFRESH_INT               15
GSLB_LICENSE_KEY                  (no valid license)
HTTP_CASE_SENSITIVE_MATCHING      1
HTTP_URL_COOKIE_DELIMITERS        /?&#+
INBAND_STATE_CHANGED_MSG_RATE     4
INFINITE_IDLE_TIME_MAXCONNS       1024
MAX_PARSE_LEN_MULTIPLIER          1
NAT_CLIENT_HASH_SOURCE_PORT       0
NO_RESET_UNIDIRECTIONAL_FLOWS     0
REAL_SLOW_START_ENABLE            3
ROUTE_UNKNOWN_FLOW_PKTS           0
SASP_CSM_UNIQUE_ID                Cisco-CSM
SASP_FIRST_BIND_ID                65520
SASP_GWM_BIND_ID_MAX              1
SASP_SCALE_WEIGHTS                0
SSL_DEFAULT_STICKY                0
SWITCHOVER_RP_ACTION              0
SWITCHOVER_SP_ACTION              0
SYN_COOKIE_INTERVAL               3
SYN_COOKIE_THRESHOLD              5000
TCP_MSS_OPTION                    1460
TCP_WND_SIZE_OPTION               8192
VSERVER_ICMP_ALWAYS_RESPOND       false
XML_CONFIG_AUTH_TYPE              Basic
MSTS_RDP_VIP_LIST                 MSTS-RDP-200 XML_TEST1
MAX_VSERVERS_PER_VIP              10
SECURE_HTTP_PORT                  443
SECURE_HTTP_SSL_METHOD            0
SECURE_HTTP_TFTP_HOST_IPADDRESS
SECURE_HTTP_SERVER_CERTIFICATE
SECURE_HTTP_PRIV_KEY_FILE
SECURE_SASP_ENABLE                0
SECURE_SASP_SSL_METHOD            0
SECURE_SASP_TFTP_HOST_IPADDRESS
SECURE_SASP_SERVER_CERTIFICATE
SECURE_SASP_PRIV_KEY_FILE

Router#
```

コンフィギュレーションの現在の環境変数セットのすべての情報を表示するには、次のように**show module csm** *slot* **variable detail** コマンドを使用します。

```
Router# show mod csm 5 variable detail
Name:ARP_INTERVAL  Rights:RW
Value:300
Default:300
Valid values:Integer (15 to 31536000)
Description:
Time (in seconds) between ARPs for configured hosts

Name:ARP_LEARNED_INTERVAL  Rights:RW
Value:14400
Default:14400
Valid values:Integer (60 to 31536000)
Description:
Time (in seconds) between ARPs for learned hosts

Name:ARP_GRATUITOUS_INTERVAL  Rights:RW
Value:15
Default:15
Valid values:Integer (10 to 31536000)
Description:
Time (in seconds) between gratuitous ARPs

Name:ARP_RATE  Rights:RW
Value:10
Default:10
Valid values:Integer (1 to 60)
Description:
Seconds between ARP retries

Name:ARP_RETRIES  Rights:RW
Value:3
Default:3
Valid values:Integer (2 to 15)
Description:
Count of ARP attempts before flagging a host as down

Name:ARP_LEARN_MODE  Rights:RW
Value:1
Default:1
Valid values:Integer (0 to 1)
Description:
Indicates whether CSM learns MAC address on responses only (0) or all traffic (1)

Name:ARP_REPLY_FOR_NO_INSERVICE_VIP  Rights:RW
Value:0
Default:0
Valid values:Integer (0 to 1)
Description:
Whether the CSM would reply to ARP for out-of-service vserver

Name:ADVERTISE_RHI_FREQ  Rights:RW
Value:10
Default:10
Valid values:Integer (1 to 65535)
Description:
The frequency in second(s) the CSM will check for RHI updates

Name:AGGREGATE_BACKUP_SF_STATE_TO_VS  Rights:RW
Value:0
Default:0
Valid values:Integer (0 to 1)
Description:
Whether to include the operational state of a backup serverfarm into the state of a
virtual server
```

```
Name:COOKIE_INSERT_EXPIRATION_DATE  Rights:RW
Value:Fri, 1 Jan 2010 01:01:50 GMT
Default:Fri, 1 Jan 2010 01:01:50 GMT
Valid values:String (2 to 63 chars)
Description:
Configuring the expiration time and date for the HTTP Cookie inserted by the CSM

Name: CSM_FAST_FIN_TIMEOUT  Rights: RW
Value: 10
Default: 10
Valid values: Integer (10 to 65535)
Description:
Timeout (in seconds) for connection reset after FIN is deteted

Name:DEST_UNREACHABLE_MASK  Rights:RW
Value:0xffff
Default:65535
Valid values:Integer (0 to 65535)
Description:
Bitmask defining which ICMP destination unreachable codes are to be forwarded

Name:FT_FLOW_REFRESH_INT  Rights:RW
Value:15
Default:15
Valid values:Integer (1 to 65535)
Description:
FT slowpath flow refresh interval in seconds

Name:GSLB_LICENSE_KEY  Rights:RW
Value:(no valid license)
Default:(no valid license)
Valid values:String (1 to 63 chars)
Description:
License key string to enable GSLB feature

Name:HTTP_CASE_SENSITIVE_MATCHING  Rights:RW
Value:1
Default:1
Valid values:Integer (0 to 1)
Description:
Whether the URL (Cookie, Header) matching and sticky to be case sensitive

Name:HTTP_URL_COOKIE_DELIMITERS  Rights:RW
Value:/?&#+
Default:/?&#+
Valid values:String (1 to 64 chars)
Description:
Configuring the list of delimiter characters for Cookies in the URL string

Name:INBAND_STATE_CHANGED_MSG_RATE  Rights:RW
Value:4
Default:4
Valid values:Integer (0 to 32)
Description:
Maximum logging messages per second when real server changed state within inband

Name:INFINITE_IDLE_TIME_MAXCONNS  Rights:RW
Value:1024
Default:1024
Valid values:Integer (1 to 4294967295)
Description:
Maximum number of connectins with infinite idle timout

Name:MAX_PARSE_LEN_MULTIPLIER  Rights:RW
Value:1
Default:1
Valid values:Integer (1 to 16)
Description:
Multiply the configured max-parse-len by this amount
```

```
Name:NAT_CLIENT_HASH_SOURCE_PORT  Rights:RW
Value:0
Default:0
Valid values:Integer (0 to 1)
Description:
Whether to use the source port to pick client NAT IP address

Name:NO_RESET_UNIDIRECTIONAL_FLOWS  Rights:RW
Value:0
Default:0
Valid values:Integer (0 to 1)
Description:
If set, unidirectional flows will not be reset when timed out

Name:REAL_SLOW_START_ENABLE  Rights:RW
Value:3
Default:3
Valid values:Integer (0 to 10)
Description:
Disable or Enable Slow Start feature with average number of conn sent to the slowstart
server. It is represented in powers of 2

Name:ROUTE_UNKNOWN_FLOW_PKTS  Rights:RW
Value:0
Default:0
Valid values:Integer (0 to 3)
Description:
Whether to route non-SYN packets that do not matched any existing flows

Name:SASP_CSM_UNIQUE_ID  Rights:RW
Value:Cisco-CSM
Default:Cisco-CSM
Valid values:String (3 to 63 chars)
Description:
Text identifier of this CSM to GWM running SASP

Name:SASP_FIRST_BIND_ID  Rights:RW
Value:65520
Default:65520
Valid values:Integer (1 to 65525)
Description:
Treat DFP bind_ids as SASP IDs starting at this value

Name:SASP_GWM_BIND_ID_MAX  Rights:RW
Value:1
Default:1
Valid values:Integer (0 to 8)
Description:
Maximum number of GWMS/bind_ids using SASP

Name:SASP_SCALE_WEIGHTS  Rights:RW
Value:0
Default:0
Valid values:Integer (0 to 12)
Description:
Scale SASP weights by N, 12 means map range to CSM range

Name:SSL_DEFAULT_STICKY  Rights:RW
Value:0
Default:0
Valid values:Integer (0 to 1)
Description:
Stick to Source IP Sticky on Unknown or BAD SSL format
```

```
Name:SWITCHOVER_RP_ACTION  Rights:RW
Value:0
Default:0
Valid values:Integer (0 to 1)
Description:
Whether to recover (0) or halt/reboot (1) after a Supervisor RP switchover occurs

Name:SWITCHOVER_SP_ACTION  Rights:RW
Value:0
Default:0
Valid values:Integer (0 to 1)
Description:
Whether to recover (0) or halt/reboot (1) after a Supervisor SP switchover occurs

Name:SYN_COOKIE_INTERVAL  Rights:RW
Value:3
Default:3
Valid values:Integer (1 to 60)
Description:
The interval, in seconds, at which a new syn-cookie key is generated

Name:SYN_COOKIE_THRESHOLD  Rights:RW
Value:5000
Default:5000
Valid values:Integer (0 to 1048576)
Description:
The threshold (in number of pending sessions) at which syn-cookie is engaged

Name:TCP_MSS_OPTION  Rights:RW
Value:1460
Default:1460
Valid values:Integer (1 to 65535)
Description:
Maximum Segment Size (MSS) value sent by CSM for L7 processing

Name:TCP_WND_SIZE_OPTION  Rights:RW
Value:8192
Default:8192
Valid values:Integer (1 to 65535)
Description:
Window Size value sent by CSM for L7 processing

Name:VSERVER_ICMP_ALWAYS_RESPOND  Rights:RW
Value:false
Default:false
Valid values:String (1 to 5 chars)
Description:
If "true" respond to ICMP probes regardless of vserver state

Name:XML_CONFIG_AUTH_TYPE  Rights:RW
Value:Basic
Default:Basic
Valid values:String (5 to 6 chars)
Description:
HTTP authentication type for xml-config:Basic or Digest

Name:MSTS_RDP_VIP_LIST  Rights:RW
Value:MSTS-RDP-200 XML_TEST1
Valid values:String (0 to 256 chars)
Description:
List of VIPs supporting MSTS-RDP Protocol

Name:MAX_VSERVERS_PER_VIP  Rights:RW
Value:10
Default:10
Valid values:Integer (7 to 10)
Description:
Configure the maximum limit for Virtual servers having the same IP address. It is
represented in powers of 2
```

```
Name:SECURE_HTTP_PORT  Rights:RW
Value:443
Default:443
Valid values:Integer (1 to 65535)
Description:
HTTPS server port number

Name:SECURE_HTTP_SSL_METHOD  Rights:RW
Value:0
Default:0
Valid values:Integer (0 to 3)
Description:
SSL version used by the HTTPS server

Name:SECURE_HTTP_TFTP_HOST_IPADDRESS  Rights:RW
Value:
Valid values:String (0 to 16 chars)
Description:
IP address of TFTP Server that contains the HTTP server Certificates

Name:SECURE_HTTP_SERVER_CERTIFICATE  Rights:RW
Value:
Valid values:String (0 to 256 chars)
Description:
Certificate file used by the HTTPS server

Name:SECURE_HTTP_PRIV_KEY_FILE  Rights:RW
Value:
Valid values:String (0 to 256 chars)
Description:
Private Key file used by the HTTPS server

Name:SECURE_SASP_ENABLE  Rights:RW
Value:0
Default:0
Valid values:Integer (0 to 1)
Description:
To enable Secure SASP

Name:SECURE_SASP_SSL_METHOD  Rights:RW
Value:0
Default:0
Valid values:Integer (0 to 3)
Description:
SSL version used by the secure SASP client

Name:SECURE_SASP_TFTP_HOST_IPADDRESS  Rights:RW
Value:
Valid values:String (0 to 16 chars)
Description:
IP address of TFTP Server that contains the SASP client Certificates

Name:SECURE_SASP_SERVER_CERTIFICATE  Rights:RW
Value:
Valid values:String (0 to 256 chars)
Description:
Certificate file used by the SASP client

Name:SECURE_SASP_PRIV_KEY_FILE  Rights:RW
Value:
Valid values:String (0 to 256 chars)
Description:
Private Key file used by the SASP client
```

連続（persistent）接続の設定

CSM では、HTTP ヘッダー内の URL、Cookie、またはその他のフィールドに基づいて HTTP 接続をスイッチングできます。CSM の連続（persistent）接続サポートにより、連続接続での後続 HTTP 要求はそれぞれ別々にスイッチング可能です。新しい HTTP 要求が届いた場合その要求は、前の要求と同じサーバにスイッチングしたり、別のサーバにスイッチングしたり、またはクライアントにリセットしてその要求が完了しないように設定することができます。

ソフトウェア Release 2.1(1) の時点で、CSM は HTTP 1.1 の連続機能（persistence）をサポートしています。この機能によって、ブラウザは 1 つの連続接続で複数の HTTP 要求を送信できます。連続接続の確立後、サーバは同じクライアントからさらに要求が届く可能性を想定して、設定可能なインターバルの間、接続をオープンな状態にしておきます。連続接続によって、要求のたびに新しい TCP 接続を確立することに伴うオーバーヘッドが排除されます。

HTTP 1.1 の連続機能はデフォルトとして、レイヤ 7 ポリシーで設定されたすべての仮想サーバでイネーブルです。連続接続をディセーブルにする場合は、**no persistent rebalance** コマンドを入力します。連続接続をイネーブルにする場合は、**persistent rebalance** コマンドを入力します。

次に、連続接続を設定する例を示します。

```
Router# configure terminal
Enter configuration commands, one per line.  End with
CNTL/Z.
Router(config)# mod csm 2
!!! configuring serverfarm
Router(config-module-csm)# serverfarm sf3
Router(config-slb-sfarm)# real 10.1.0.105
Router(config-slb-real)# inservice
!!! configuring vserver
Router(config-slb-real)# vserver vs3
Router(config-slb-vserver)# virtual 10.1.0.83 tcp 80
Router(config-slb-vserver)# persistent rebalance
Router(config-slb-vserver)# serverfarm sf3
Router(config-slb-vserver)# inservice
Router(config-slb-vserver)# end
```

HTTP ヘッダー挿入

HTTP のヘッダー挿入は、CSM にクライアント IP アドレスの HTTP ヘッダーへの挿入のような情報を挿入させる機能です。この機能は、CSM が送信元 Network Address Translation（NAT; ネットワーク アドレス変換）を実行していて、さらにサーバ側のアプリケーションがまだ最初の送信元 IP を要求している場合に特に有効です。

CSM は、クライアントからサーバ方向のヘッダーに、クライアントからの送信元 IP アドレスを挿入できます。

HTTP ヘッダーに情報を挿入するには、**insert protocol http header** *name* **header-value** *value* コマンドを使用します。

- *name* ― HTTP ヘッダーの一般フィールドの文字どおりの名前。名前は 1 ～ 63 文字のストリングです。
- *value* ― 要求に挿入するヘッダー値のストリングを文字どおり指定します。

  また、ヘッダー値の特殊パラメータ %is および %id を使用できます。%is 値は HTTP ヘッダーに送信元 IP アドレスを挿入し、%id 値は HTTP ヘッダーに宛先 IP アドレスを挿入します。特殊パラメータはそれぞれヘッダー マップごとに 1 度ずつ指定できます。

（注） ヘッダー マップには、複数の挿入ヘッダーが含まれることがあります。スペースを含む複数のキーワードからなるヘッダー値を挿入する場合、全体を二重引用符で囲む必要があります。

HTTP ヘッダー挿入を設定する場合、ヘッダー マップおよびポリシーを使用する必要があります。HTTP ヘッダー挿入を機能させるには、デフォルトのポリシーは適用できません。

次に、ヘッダー フィールドと値を指定して、要求を検索する例を示します。

```
Router(config-module-csm)# natpool TESTPOOL 10.10.110.200 10.10.110.210 netmask
255.255.255.0
!
Router(config-module-csm)# map HEADER-INSERT header
Router(config-slb-map-header)# insert protocol http header Source-IP header-value %is
Router(config-slb-map-header)# insert protocol http header User-Agent header-value
"MyBrowser 1.0"
!
Router(config-module-csm)# real SERVER1
Router(config-slb-real)# address 10.10.110.10
Router(config-slb-real)# inservice
Router(config-module-csm)# real SERVER2
Router(config-slb-real)# address 10.10.110.20
Router(config-slb-real)# inservice
!
Router(config-module-csm)# serverfarm FARM-B
Router(config-slb-sfarm)# nat server
Router(config-slb-sfarm)# nat client TESTPOOL
Router(config-slb-real)# real name SERVER1
Router(config-slb-real)# inservice
Router(config-slb-real)# real name SERVER2
Router(config-slb-real)# inservice
!
Router(config-module-csm)# policy INSERT
Router(config-slb-policy)# header-map HEADER-INSERT
Router(config-slb-policy)# serverfarm FARM-B
!
Router(config-module-csm)# vserver WEB
Router(config-slb-vserver)# virtual 10.10.111.100 tcp www
Router(config-slb-vserver)# persistent rebalance
Router(config-slb-vserver)# slb-policy INSERT
Router(config-slb-vserver)# inservice
```

GSLB の設定

ここでは、CSM Global Server Load Balancing（GSLB; グローバル サーバ ロードバランシング）の拡張機能セット オプション、およびその使用方法について説明します。拡張機能セット オプションを使用する前に、「はじめに」の「ライセンス」（p.xxiv）および表紙裏に記載されているソフトウェア使用許諾書の諸条件を入念に確認してください。

（注）　ソフトウェアをダウンロードまたはインストールすることにより、使用許諾書に同意することになります。この許諾書のすべての条項に同意できない場合は、ソフトウェアをダウンロード、インストール、または使用しないでください。

GSLB 拡張機能セット オプションの使用

GSLB をイネーブルにするには、イネーブル モードで次の手順を実行します。

コマンド	目的
`Router# config t` `Router(config)# mod csm 5`	コンフィギュレーション モードを開始して、特定の CSM（たとえば、ここで使用されているモジュール 5 など）の CSM コンフィギュレーション モードを開始します。
`Router(config-module-csm)# variable` *name value*	次のように名前と値を指定して、GSLB をイネーブルにします。[1] • Name= • Value=
`Router(config-module-csm)# exit` `Router (config)# write mem`	CSM モジュール コンフィギュレーション モードを終了して、設定の変更を保存します。
`Router#:hw-module` *slot number* `reset`	CSM を再起動して、変更をアクティブにします。

1. GSLB では、別途ライセンスを購入する必要があります。ご使用の GSLB のライセンスを購入する場合は、製品を購入された代理店にご連絡ください。

表 8-2 に、CSM で使用する GSLB の環境変数の値を示します。

表 8-2　GSLB 環境変数の値

名前	デフォルト	有効値	説明
GSLB_LICENSE_KEY	（有効なライセンスはありません）	ストリング（1 ～ 63 文字）	GSLB 機能をイネーブルにするライセンス キーのストリングを指定します。
GSLB_KALAP_UDP_PORT	5002	整数（1 ～ 65535）	GSLB KAL-AP UDP のポート番号を指定します。
GSLB_KALAP_PROBE_FREQ	45	整数（45 ～ 65535）	GSLB KAL-AP プローブの頻度を指定します。
GSLB_KALAP_PROBE_RETRIES	3	整数（1 ～ 65535）	GSLB KAL-AP プローブの最大再試行回数を指定します。
GSLB_ICMP_PROBE_FREQ	45	整数（45 ～ 65535）	GSLB ICMP プローブの頻度を指定します。
GSLB_ICMP_PROBE_RETRIES	3	整数（1 ～ 65535）	GSLB ICMPプローブの最大再試行回数を指定します。
GSLB_HTTP_PROBE_FREQ	45	整数（45 ～ 65535）	GSLB HTTP プローブの頻度を指定します。
GSLB_HTTP_PROBE_RETRIES	3	整数（1 ～ 65535）	GSLB HTTP プローブの最大再試行回数を指定します。

表 8-2　GSLB 環境変数の値（続き）

名前	デフォルト	有効値	説明
GSLB_DNS_PROBE_FREQ	45	整数（45 ～ 65535）	GSLB DNS プローブの頻度を指定します。
GSLB_DNS_PROBE_RETRIES	3	整数（1 ～ 65535）	GSLB DNS プローブの最大再試行回数を指定します。

GSLB の設定

GSLB は、利用可能な負荷に基づき、Domain Name Server（DNS; ドメイン ネーム サーバ）を通してさまざまなサーバ ファームおよび実サーバにクライアント接続を振り分けることによって、散在している複数のホスト サイト間で負荷を分散させます。GSLB は、アクセス リスト、マップ、サーバ ファーム、およびロードバランス アルゴリズムを使用して実行されます。表 8-3 に、CSM 上で GSLB を設定するための要件をまとめます。

表 8-3　GSLB の動作

クライアント要求（起点）	ドメイン（目的）	サーバ ファーム（終点）	アルゴリズム（方法）
着信DNS要求のフィルタリングには、アクセス リストを使用できます。着信 DNS 要求に設定されているマップ、クライアント グループ、およびサーバ ファームを関連付けるには、ポリシーを使用します。	マップを設定して、クライアント要求と一致しなければならないドメイン名を指定します。正規表現の構文を使用できます。 たとえば、クライアント要求は cnn.com や yahoo.com などのドメイン名が一致している必要があります。ドメイン名が指定したポリシー マップに一致する場合は、要求に応答するための実サーバについてプライマリ サーバ ファームが照会されます。	サーバ ファームでは、クライアント要求を満たす情報の検索先として、実サーバ グループを指定します。	ターゲット実サーバのアベイラビリティを判別するために、GSLB プローブを使用できます。実サーバに設定されているプローブ タイプを使用します。 GSLBサーバ ファーム プレディクタは、ラウンドロビン式の最小負荷、順序付きリスト、ハッシュ アドレス ソース、ハッシュ ドメイン、およびハッシュ ドメイン アドレス ソースです。

図 8-1 に、GSLB の基本的な設定を示します。

図 8-1　GSLB の設定

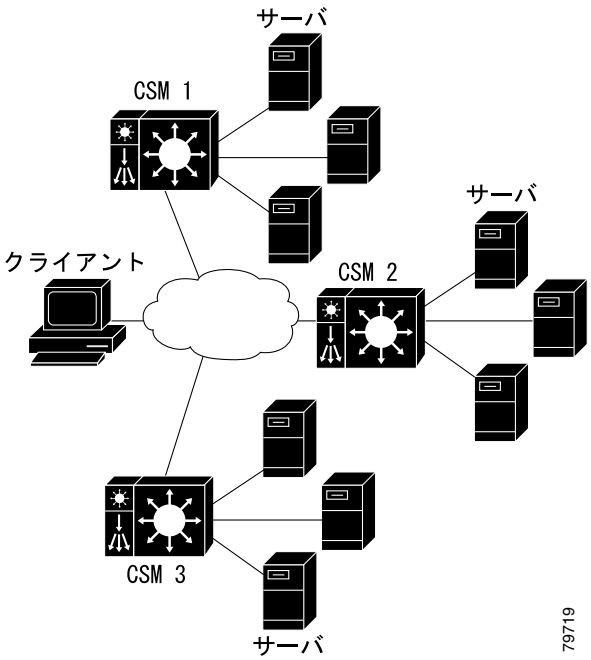


図 8-1 の場合、設定作業と例に関する注意事項は次のとおりです。

- CSM 1 は GSLB および SLB の両方を実行しますが、CSM 2 および CSM 3 が実行するのは SLB だけです。
- CSM 1 には、SLB 用の仮想サーバと GSLB 用の仮想サーバがあります。SLB 用の仮想サーバでは、サーバ ファームの実サーバはローカル サーバの IP アドレスです。
- DNS ポリシーではプライマリ サーバ ファームを使用します。実サーバの 1 つがローカルで、ほかの 2 つの実サーバは、それぞれ CSM 2 および CSM 3 上で設定された仮想サーバです。
- 両方のリモート ロケーション、ローカル実サーバ、および仮想サーバにプローブを追加する必要があります。
- CSM 1 の管理用 IP アドレス（CSM 1 の VLAN アドレスまたはエイリアス IP）に送信された DNS 要求には、GSLBFARM というサーバ ファームで設定された 3 つの実サーバ IP アドレスのうちの 1 つが応答として与えられます。

GSLB を設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-slb-vserver)# serverfarm serverfarm-name`	仮想サーバに関連付けるサーバ ファームを作成します。
ステップ 2	`Router(config-module-csm)# vserver virtserver-name`	CSM 1 上で SLB 用の仮想サーバを指定し、仮想サーバ サブモードを開始します。
ステップ 3	`Router(config-slb-vserver)# virtual ip-address [ip-mask] protocol port-number [service ftp]`	仮想サーバの属性を設定します。
ステップ 4	`Router(config-slb-vserver)# inservice`	仮想サーバのロードバランシングをイネーブルにします。

	コマンド	目的
ステップ 5	`Router(config-module-csm)# vserver virtserver-name dns`	GSLB 用の仮想サーバを指定し、仮想サーバ サブモードを開始します。
ステップ 6	`Router(config-slb-vserver)# dns-policy [group group-id] [netmask ip-netmask]`	同じクライアントからの接続には同じサーバ ファームが使用されるようにします。
ステップ 7	`Router(config-slb-vserver)# inservice`	仮想サーバの GSLB をイネーブルにします。
ステップ 8	`Router(config-module-csm)# serverfarm GSLBFARM dns-vip`	GSLBFARM サーバ ファーム（実際にはフォワーディング ポリシー）を作成して名前を指定し、サーバ ファーム コンフィギュレーション モードを開始します。
ステップ 9	`Router(config-slb-sfarm)# predictor hash address source`	サーバ ファームにロードバランス プレディクタのハッシュ アドレス ソースを設定します。
ステップ 10	`Router(config-module-csm)# real ip-addres`	実サーバのエイリアス IP アドレスを指定し、実サーバ コンフィギュレーション サブモードを開始します。
ステップ 11	`Router(config-slb-real)# inservice`	仮想サーバのロードバランシングをイネーブルにします。
ステップ 12	`Router(config-module-csm)#map dns-map-name dns`	DNS マップを設定します。
ステップ 13	`Router(config-dns-map)# match protocol dns domain name`	DNS マップに DNS 名を追加します。
ステップ 14	`Router(config-module-csm)#policy policy name`	ポリシーを設定します。
ステップ 15	`Router(config-slb-policy)# dns map map_name`	ポリシーに DNS マップ属性を追加します。
ステップ 16	`Router(config-slb-policy)#serverfarm primary-serverfarm [backup sorry-serverfarm [sticky]]`	サーバ ファームとポリシーを関連付けます。
ステップ 17	`Router(config-module-csm)# vserver virtserver-name`	CSM 2 上で仮想サーバを設定し、仮想サーバ サブモードを開始します。
ステップ 18	`Router(config-slb-vserver)# virtual ip-address [ip-mask] protocol port-number [service ftp]`	仮想サーバの属性を設定します。
ステップ 19	`Router(config-slb-vserver)# serverfarm serverfarm-name`	サーバ ファームと仮想サーバを関連付けます。
ステップ 20	`Router(config-slb-vserver)# inservice`	仮想サーバのロードバランシングをイネーブルにします。
ステップ 21	`Router(config-module-csm)# vserver virtserver-name`	CSM 3 上で仮想サーバを設定し、仮想サーバ サブモードを開始します。
ステップ 22	`Router(config-slb-vserver)# virtual ip-address [ip-mask] protocol port-number [service ftp]`	仮想サーバの属性を設定します。
ステップ 23	`Router(config-slb-vserver)# serverfarm serverfarm-name`	サーバ ファームと仮想サーバを関連付けます。
ステップ 24	`Router(config-slb-vserver)# inservice`	仮想サーバのロードバランシングをイネーブルにします。

次に、GSLB を設定する例を示します。

CSM 1 上で：

```
Router(config-module-csm)# serverfarm WEBFARM
Router(config-slb-sfarm)# predictor round-robin
Router(config-slb-sfarm)# real 3.5.5.5
Router(config-slb-real)# inservice
Router(config-slb-sfarm)# real 3.5.5.6
Router(config-slb-real)# inservice
Router(config-slb-real)# exit
Router(config-slb-sfarm)# exit

Router(config-module-csm)# vserver WEB
Router(config-slb-vserver)# virtual 10.10.10.10 tcp www
Router(config-slb-vserver)# serverfarm WEBFARM
Router(config-slb-vserver)# inservice

Router(config-module-csm)# serverfarm GSLBSERVERFARM dns-vip
Router(config-slb-sfarm)# predictor round-robine
Router(config-slb-sfarm)# real 10.10.10.10
Router(config-slb-real)# inservice
Router(config-slb-real)# exit
Router(config-slb-sfarm)# real 20.20.20.20
Router(config-slb-real)# inservice
Router(config-slb-real)# exit
Router(config-slb-sfarm)# real 30.30.30.30
Router(config-slb-real)# inservice
Router(config-slb-real)# exit

Router(config-module-csm)# map MAP1 dns
Router(config-dns-map)# match protocol dns domain foobar.com
Router(config-dns-map)# exit

Router(config-module-csm)# policy DNSPOLICY dns
Router(config-slb-policy)# dns map MAP1
Router(config-slb-policy)# serverfarm primary GSLBSERVERFARM ttl 20 responses 1
Router(config-slb-policy)# exit

Router(config-module-csm)# vserver DNSVSERVER dns
Router(config-slb-vserver)# dns-policy DNSPOLICY
Router(config-slb-vserver)# inservice
```

CSM 2 上で：

```
Router(config-module-csm)# serverfarm WEBFARM
Router(config-slb-sfarm)# predictor round-robin
Router(config-slb-sfarm)# real 4.5.5.5
Router(config-slb-real)# inservice
Router(config-slb-sfarm)# real 4.5.5.6
Router(config-slb-real)# inservice
Router(config-slb-real)# exit
Router(config-slb-sfarm)# exit

Router(config-module-csm)# vserver WEB
Router(config-slb-vserver)# virtual 20.20.20.20 tcp www
Router(config-slb-vserver)#s erverfarm WEBFARM
Router(config-slb-vserver)# inservice
```

CSM 3 上で：

```
Router(config-module-csm)# serverfarm WEBFARM
Router(config-slb-sfarm)# predictor round-robin
Router(config-slb-sfarm)# real 5.5.5.5
Router(config-slb-real)# inservice
Router(config-slb-sfarm)# real 5.5.5.6
Router(config-slb-real)# inservice
Router(config-slb-real)# exit
Router(config-slb-sfarm)# exit
Router(config-module-csm)# vserver WEB
Router(config-slb-vserver)# virtual 30.30.30.30 tcp www
Router(config-slb-vserver)# serverfarm WEBFARM
Router(config-slb-vserver)# inservice
```

ネットワーク管理の設定

ここでは、ネットワーク上での CSM の管理方法について説明します。

実サーバの SNMP トラップの設定

SNMP トラップはイネーブルの場合、実サーバのステートが変わるたびに（たとえば、サーバがサービスを開始または停止するたびに）外部の管理装置に送信されます。トラップには、実サーバトラップであることを示す Object Identifier（OID; オブジェクト識別子）が含まれます。

（注）　実サーバ トラップの OID は 1.3.6.1.4.1.9.9.161.2 です。

トラップには、サーバ ステートが変わった理由を示すメッセージも含まれます。

Catalyst 6500 シリーズ スイッチの SLB 機能に関連付けられたフォールトトレラント トラップをイネーブルまたはディセーブルにするには、**snmp-server enable traps slb ft** コマンドを使用します。フォールトトレラント トラップは、SLB のフォールトトレラントの要素を扱います。たとえば、フォールトトレラント トラップがイネーブルで、SLB 装置がフォールトトレラント ピアの障害を検出した場合、その SLB 装置はスタンバイからアクティブに変わるときに、SNMP トラップを送信します。

実サーバ用の SNMP トラップを設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router (config)# snmp-server community public`	通知動作で送信される、パスワードと同様のコミュニティ ストリングを定義します。**public**はその一例です。
ステップ 2	`Router (config)# snmp-server host host-addr`	トラップの送信先となる、外部ネットワーク管理装置の IP アドレスを定義します。
ステップ 3	`Router (config)# snmp-server enable traps slb csrp`	実サーバ用の SNMP トラップをイネーブルにします[1]。

1. SNMP フォールトトレラント トラップ機能をディセーブルにする場合は、このコマンドの **no** 形式を使用します。

XML インターフェイスの設定

従来のリリースでは、Cisco IOS CLI（コマンドライン インターフェイス）が CSM を設定する唯一の手段でした。XML により、Document Type Definition（DTD）を使用して CSM を設定できます。XML DTD の例については、付録 C「CSM XML の DTD」を参照してください。

CSM で XML を使用する場合、注意事項は次のとおりです。

- 同時に使用できるクライアント接続は最大で 5 つです。
- XML の設定は IP SLB モードとは無関係ですが、**csm_module slot='x' sense='no'** コマンドの場合は例外的に所定の結果をもたらし、XML エラーを生成します。
- パイプラインの HTTP POST はサポートされません。
- すべてのクライアント通信が 30 秒でタイムアウトします。
- クライアント証明書が不良だった場合、Cisco IOS のシステム ログにメッセージが送信されます。
- 異なるスロット属性を指定することによって、1 つの CSM をほかの CSM コンフィギュレーションのプロキシにすることができます。

この機能をイネーブルにすると、ネットワーク管理装置が CSM に接続し、新しい設定を装置に送信する場合があります。ネットワーク管理装置は、標準の HTTP プロトコルを使用して、コンフィギュレーション コマンドを CSM に送信します。HTTP POST のデータ部分で、XML 文書を CSM に送信することによって、新しい設定が適用されます。

HTTP 会話の例を示します。

```
******** Client **************
POST /xml-config HTTP/1.1
Authorization: Basic VTpQ
Content-Length: 95

<?xml version="1.0"?>
<config><csm_module slot="4"><vserver name="FOO"/></csm_module></config>
******** Server **************
HTTP/1.1 200 OK
Content-Length: 21

<?xml version="1.0"?>
******** Client **************
POST /xml-config HTTP/1.1
Content-Length: 95

<?xml version="1.0"?>
<config><csm_module slot="4"><vserver name="FOO"/></csm_module></config>
******** Server **************
HTTP/1.1 401 Unauthorized
Connection: close
WWW-Authenticate: Basic realm=/xml-config
```

表 8-4 に、サポート対象の HTTP 戻りコードを示します。

表 8-4　XML に関する HTTP の戻りコード

戻りコード	説明
200	OK
400	不良要求
401	未承認（要求した証明書が提出されなかった）
403	禁止（無効な証明書が提出され、Syslog も生成された）
404	未検出（「/xml-config」が指定されていない）
408	要求のタイムアウト（受信待ちで 30 秒以上経過）
411	コンテンツ長の脱落（Content-Length フィールドが脱落またはゼロ）
500	内部サーバ エラー
501	実装されていない（POST が指定されていない）
505	サポートされない HTTP バージョン（1.0 または 1.1 が指定されていない）

次の HTTP ヘッダーがサポートされます。

- Content-Length（すべての POST にゼロ以外の値が必要）
- Connection（*close* 値は要求を連続 [persistent] させないことを指定）
- WWW-Authenticate（要求した証明書がない場合にクライアントに送信）
- Authorization（base64 符号化方式による基本証明書を指定するためにクライアントから送信）

XML 機能を動作させるには、ネットワーク管理システムがスイッチ インターフェイスの IP アドレスではなく、CSM の IP アドレスに接続する必要があります。

コマンドライン インターフェイスの場合と同様、コンフィギュレーションのマスター コピーを Cisco IOS ソフトウェアに保存しなければならないので、XML コンフィギュレーション要求を受信した CSM は、これらの要求をスーパーバイザ エンジンに送らなければなりません。

（注） XML コンフィギュレーションによって、1 つの CSM を同一スイッチ シャーシに搭載されたすべての CSM のプロキシとして動作させることができます。たとえば、ある CSM 用のコンフィギュレーションが含まれる XML ページを、同じスイッチ シャーシに搭載された別の CSM から正しく提供できます。

現在、一般公開されている DTD は、作成する XML コンフィギュレーション文書の基盤です（付録 C「CSM XML の DTD」を参照）。XML 文書は HTTP POST 要求によって、CSM に直接送られます。XML を使用するには、Cisco IOS の CLI を使用して、前もって CSM 上で最小限のコンフィギュレーションを作成しておく必要があります。**xml-config** コマンドについては、『*Catalyst 6500 Series Content Switching Module Command Reference*』を参照してください。

応答は要求をミラー化した XML 文書です。問題のある要素にはチャイルド エラー要素でフラグが設定され、エラー コードおよびエラー文字列が示されます。XML 文書でルート要素の属性を使用することによって、無視すべきエラーのタイプを指定できます。

Cisco IOS CLI には、特定の CSM インターフェイスの XML コンフィギュレーション機能をイネーブルにするコマンドが追加される予定です。イネーブル / ディセーブル機能とともに、TCP ポート、クライアント アクセス リストのセキュリティ オプション、および HTTP 認証がサポートされます。

CSM 上で XML を設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# module csm slot`	モジュールおよびスロット番号を指定します。
ステップ 2	`Router(config-module-csm)# xml-config`	CSM 上で XML をイネーブルにして、XML コンフィギュレーション モードを開始します。
ステップ 3	`Router(config-slb-xml)# port port-number`	CSM HTTP サーバが待ち受ける TCP ポートを指定します。
ステップ 4	`Router(config-slb-xml)# vlan id`	CSM HTTP サーバが指定された VLAN からの接続だけを受け付けるように制限します。
ステップ 5	`Router(config-slb-xml)# client-group [1-99 | name]`	CSM XML コンフィギュレーション インターフェイスが受け付けるのは、クライアント グループと一致する IP アドレスからの接続だけであることを指定します。
ステップ 6	`Router(config-slb-xml)# credentials user-name password`	ユーザ名とパスワードのコンビネーションを 1 つまたは複数設定します。**credentials** コマンドを 1 つまたは複数設定した場合、CSM HTTP サーバは RFC 2617 で規定された基本認証方式を使用して、ユーザ アクセスを認証します。
ステップ 7	`Router# show module csm 4 xml stats`	XML 統計情報のリストを表示します。 **（注）** 統計情報のカウンタは 32 ビットです。

次に、CSM 上で XML を設定する例を示します。

```
Router(config-module-csm)# configure terminal
Router(config-module-csm)# m odule csm 4
Router(config-module-csm)# xml-config
Router(config-slb-xml)# port 23
Router(config-slb-xml)# vlan 200
Router(config-slb-xml)# client-group 60
Router(config-slb-xml)# credentials eric @#$#%%@
Router# show module csm 4 xml stats
```

許容できない XML エラーが発生した場合は、HTTP 応答に 200 というコードが含まれます。エラーが発生した元の XML 文書の一部が、エラー タイプと説明を示したエラー要素とともに戻されます。

仮想サーバ名が脱落している場合のエラー応答の例を示します。

```
      <?xml version="1.0"?>
      <config>
       <csm_module slot="4">
        <vserver>
         <error code="0x20">Missing attribute name in element
vserver</error>
        </vserver>
       </csm_module>
      </config>
```

戻されるエラー コードは、コンフィギュレーション要素のエラー許容属性のビットとも対応しています。戻される XML エラー コードは、次のとおりです。

```
XML_ERR_INTERNAL          = 0x0001,
XML_ERR_COMM_FAILURE      = 0x0002,
XML_ERR_WELLFORMEDNESS    = 0x0004,
XML_ERR_ATTR_UNRECOGNIZED = 0x0008,
XML_ERR_ATTR_INVALID      = 0x0010,
XML_ERR_ATTR_MISSING      = 0x0020,
XML_ERR_ELEM_UNRECOGNIZED = 0x0040,
XML_ERR_ELEM_INVALID      = 0x0080,
XML_ERR_ELEM_MISSING      = 0x0100,
XML_ERR_ELEM_CONTEXT      = 0x0200,
XML_ERR_IOS_PARSER        = 0x0400,
XML_ERR_IOS_MODULE_IN_USE = 0x0800,
XML_ERR_IOS_WRONG_MODULE  = 0x1000,
XML_ERR_IOS_CONFIG        = 0x2000
```

デフォルトの error_tolerance 値は 0x48 です。これは、認識されない属性および要素の無視と対応しています。

SASP の設定

SASP によって、CSM は Workload Manager（WM）のレジスタからトラフィック ウェイトに関する推奨を受けることができます。さらに、WM から CSM に新しいロードバランシング グループ メンバーを推奨できます。

SASP は、Cisco IOS Release 12.1(13)E3 以降のリリースでサポートされます。また、4.1.2 以降のリリースをサポートする Cisco IOS リリースが必要です。

SASP を設定するには、サーバ ファーム（SASP グループなど）、および WM（SASP Global Workload Manager [GWM] など）の代理になっている DFP エージェントに特殊な bind_id を関連付ける必要があります。

SASP グループの設定

SASP グループは、CSM 上のサーバ グループに相当します。グループを設定するには、**serverfarm** コンフィギュレーション コマンドを使用します。グループ メンバーはすべて、サーバ ファームに所属するものとして設定された実サーバです。このグループを GWM に関連付けるには、GWM と一致する SASP bind_id を割り当てます。

SASP グループを設定するには、次のように、サーバ ファーム コンフィギュレーション サブメニューから **bindid** コマンドを使用します。

```
Router(config-slb-sfarm)# bindid 7
```

GWM の設定

GWM は DFP エージェントとして設定します。GWM を設定するには、CSM コンフィギュレーション コマンドから dfp サブメニューを開始する必要があります。次に、DFP エージェントとして GWM を設定する例を示します。

```
Router(config-slb-dfp)# agent ip.address port bind_id
```

（注） CLI から bind_id を入力することはできません。ただし、このエージェントを GWM として設定するには、bind_id が必須です。CLI では、bind_id キーワードを「アクティビティ タイムアウト」または「キープアライブ」として記述します。さらに 2 つの値を追加できます。ただし、SASP 環境のトラブルシューティング時を除き、追加の値は入力しないでください。

代わりに、GWM は次のように設定できます。

```
Router(config-slb-dfp)# agent ip.address port bind_id flags
```

または

```
Router(config-slb-dfp)# agent ip.address port bind_id flags keep-alive-interval
```

キープアライブ インターバルは秒数です。デフォルトは 180 です。フラグは CSM が GWM にどのように登録するかを制御します。デフォルト値はゼロです。

（注）　このコンフィギュレーションは主にデバッグに使用されるため、フラグはゼロのままにしておくことを推奨します。また、CSM はメンバーが開始したアクション（信頼）をサポートしません。したがって、値 34、35、38、および 39 はサポートされていません。

フラグの意味については、表 8-5 を参照してください。

表 8-5　SASP フラグ

フラグ値	意味
0	CSM のデフォルトの登録フラグ（37）を使用
32	GWM のデフォルト ロードバランシング登録を指定。ロードバランサは「Get Weights」メッセージを送信して新しいウェイトを取得し、GWM からそのウェイトを引き出します（pull）。 GWM には、このロードバランサにウェイトを送信するときに、（ウェイトが変わらないメンバーを含めて）すべてのグループ メンバーのウェイトを組み込む必要があります。
33	ロードバランサが「Send Weights」メッセージを介してウェイトを受信することを指定（GWM はロードバランサにウェイトを格納 [push]）
34	メンバーが開始した登録 / 登録解消を GWM に信頼させ、送信されたウェイトで登録 / 登録解消をただちに更新させます。
35	33 および 34 と同じ
36	GWMが前回の期間からウェイトが変化していないメンバーを含めてはならないことを指定
37	33 および 36 と同じ
38	34 および 36 と同じ
39	33、34、および 36 と同じ

代替 bind_id の設定

デフォルトでは、1 つの bind_id が SASP bind_id（65520）として設定されます。ただし、一定範囲の連続した bind_id を使用できます。範囲内の最初の bind_id は、1 ～ 65525 までの任意の値です。次に、ある範囲の最初の bind_id 値を設定する例を示します。

```
Router(config-module-csm)# variable SASP_FIRST_BIND_ID value
```

SASP で使用できる bind_id の最大数は 8 です。これは、サポートされる GWM の最大数でもあります。bind_id の最大数は 0 ～ 8 の任意の値に設定できます。次に、bind_id 値を設定する例を示します。

```
Router(config-module-csm)# variable SASP_GWM_BIND_ID_MAX value
```

2 つの変数を使用して、次のように設定できます。

```
variable SASP_FIRST_BIND_ID 12
variable SASP_GWM_BIND_ID_MAX 3
```

つまり、bind_id 12、13、14 を使用して 3 つの異なる GWM を設定できます。

（注）　これらの環境変数を 1 つでも変更した場合は、CSM を再起動してください。

CSM 固有の ID 設定

CSM にはデフォルトで、「Cisco-CSM」という固有の識別ストリングが与えられます。次に、CSM のコンフィギュレーション コマンドを使用して、このストリングを設定する例を示します。

```
Router(config-module-csm)# variable SASP_CSM_UNIQUE_ID text
```

（注）　これらの環境変数を 1 つでも変更した場合は、CSM を再起動してください。

ウェイト スケーリングの設定

CSM 上の実サーバのウェイトは 0 ～ 100 です。メンバーの SASP ウェイトは 0 ～ 65536 です。GWM が CSM の範囲内でウェイトを作成するかぎり、スケーリングは不要です。GWM が SASP の全範囲を使用する場合は、この範囲をマッピングする必要があります。次に、SASP ウェイトのスケーリング例を示します。

```
Router(config-module-csm)# variable SASP_SCALE_WEIGHTS value
```

SASP_SCALE_WEIGHTS の範囲は 0 ～ 12 です。0 ～ 11 の値を指定すると、SASP ウェイトが 2 の n 乗で除算されます。値 12 を指定すると、65536 の値全体が CSM の 0 ～ 100 のウェイト範囲にマッピングされます。

次に、SASP GWM の詳細の表示例を示します。

```
Router# show module csm 3 dfp detail
DFP Agent 64.100.235.159:3860  Connection state: Connected
   Keepalive = 65521  Retry Count = 33     Interval = 180   (Default)
   Security errors = 0
   Last message received: 03:33:46 UTC 01/01/70
   Last reported Real weights for Protocol any, Port 0
      Host 10.9.10.22       Bind ID 65521  Weight 71
      Host 10.10.12.10      Bind ID 65521  Weight 70
      Host 10.10.12.12      Bind ID 65521  Weight 68
   Last reported Real weights for Protocol any, Port 44
      Host 10.9.10.9        Bind ID 65521  Weight 69
DFP manager listen port not configured
No weights to report to managers
```

次に、SASP グループの表示例を示します。

```
Router# show module csm 3 serverfarms detail
SVRFARM2, type = SLB, predictor = RoundRobin, nat = SERVER
  virtuals inservice: 0, reals = 4, bind_id = 65521, fail action = none
  inband health config: <none>
  retcode map = <none>
  Real servers:
    10.10.12.10, weight = 78, OUTOFSERVICE, conns = 0
    10.10.12.12, weight = 76, OPERATIONAL, conns = 0
    10.9.10.9:44, weight = 77, OPERATIONAL, conns = 0
    10.9.10.22, weight = 79, OUTOFSERVICE, conns = 0
  Total connections = 0
```

次に、SASP 環境変数の表示例を示します。

```
Router# show module csm 3 variable
variable                        value
------------------------------------------------------------------
ARP_INTERVAL                    300
Ö
ROUTE_UNKNOWN_FLOW_PKTS         0
SASP_FIRST_BIND_ID              65520
SASP_GWM_BIND_ID_MAX            2
SASP_CSM_UNIQUE_ID              paula jones
Ö

XML_CONFIG_AUTH_TYPE            Basic
```

バックエンドの暗号化

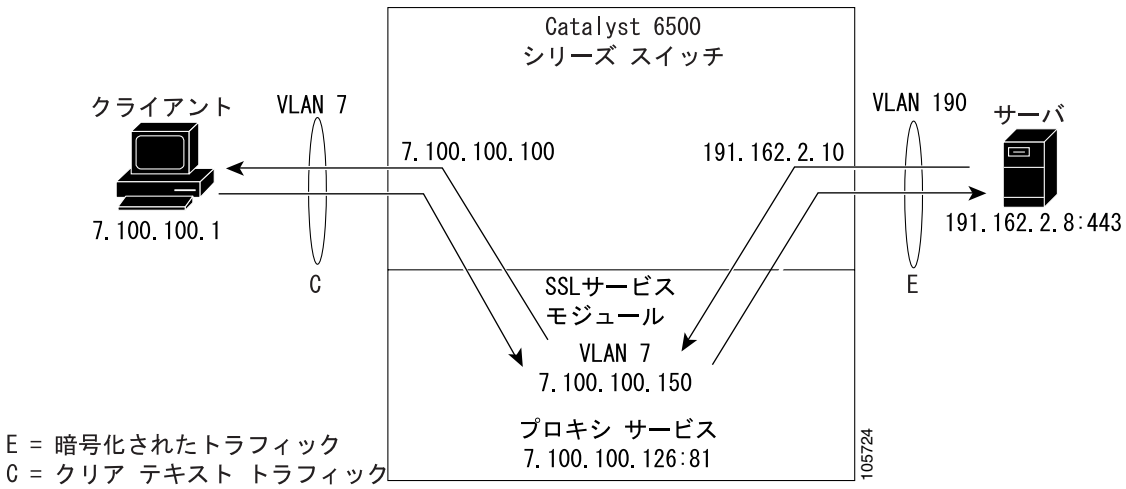
バックエンドの暗号化によって、安全なエンドツーエンド環境が実現します。図 8-2 では、クライアント（7.100.100.1）はスイッチ ポート 6/47 に接続して VLAN 7 にアクセスします。サーバ（191.162.2.8）は、スイッチ ポート 10/2 に接続して VLAN 190 にアクセスします。

SSL プロキシである VLAN7 の設定は、次のとおりです。

- IP アドレス ― 7.100.100.150
- スタティック ルートおよびゲートウェイ：
  - ― ルート 191.0.0.0
  - ― ゲートウェイ 7.100.100.100

  ゲートウェイの IP アドレス（MSFC 上の VLAN 7 の IP アドレス）は、未知のネットワークを宛先とするクライアント側のトラフィックがこのアドレスに転送され、そこからクライアントにルーティングされるように設定されています。
- クライアント側ゲートウェイ ― 7.100.100.100（MSFC 上で設定された VLAN 7 の IP アドレス）
- クライアント プロキシ サービスの仮想 IP アドレス ― 7.100.100.150.81
- サーバの IP アドレス ― 191.162.2.8

図 8-2　基本的なバックエンド暗号化



クライアント側の設定

次に、SSL プロキシ サービスの設定例を示します。

```
ssl-proxy(config)# ssl-proxy service S1
ssl-proxy(config-ssl-proxy)# virtual ipaddr 10.1.0.21 protocol tcp port 443 secondary
ssl-proxy(config-ssl-proxy)# server ipaddr 10.2.0.100 protocol TCP port 80
ssl-proxy(config-ssl-proxy)# inservice
```

次に、CSM 仮想サーバの設定例を示します。

```
Router(config-module-csm)# serverfarm SSLfarm
Router(config-slb-sfarm)# real 10.1.0.21 local
Router(config-slb-real)# inservice

Router(config-module-csm)# vserver VS1
Router(config-slb-vserver)# virtual 10.1.0.21 tcp https
Router(config-slb-vserver)# serverfarm SSLfarm
Router(config-slb-vserver)# inservice
```

SSL ロードバランシングは、混在モードの CSM および SSL Services Module 上で実行できます。

CSM は SSL-ID 固定（sticky）機能を使用して、同じ SSL Services Module に SSL 接続を固定します。CSM は、SSL-ID を調べるためにクライアント側の TCP 接続を終端させなければなりません。CSM はさらに、ロードバランシングが決定された時点で、SSL Services Module（SSLSM）への TCP 接続を開始しなければなりません。

トラフィック フローには、仮想サーバで受信したあらゆるトラフィックを SSLSM 上で終端する TCP で SSLSM に渡す CSM が含まれます。SSL 固定（sticky）機能をイネーブルにすると、CSM と SSLSM 間の接続が完全な TCP 接続になります。

次に、混在モードの SSL ロードバランシングを設定する例を示します。

```
Router(config-module-csm)# sticky 10 ssl timeout 60
Router(config-module-csm)# serverfarm SSLfarm
Router(config-slb-sfarm)# real 10.1.0.21 local
Router(config-slb-sfarm)# inservice
Router(config-slb-sfarm)# real 10.2.0.21
Router(config-slb-sfarm)# inservice
Router(config-module-csm)# vserver VS1
Router(config-slb-vserver)# virtual 10.1.0.21 tcp https
Router(config-slb-vserver)# sticky 60 group 10
Router(config-slb-vserver)# serverfarm SSLfarm
Router(config-slb-vserver)# persistent rebalance
Router(config-slb-vserver)# inservice
```

CSM がクライアント側の TCP 接続を終端させなければならないときに、SSLSM でトラフィックを転送するコンフィギュレーションを内部生成の形で作成する必要があります。サーバ ファーム *SSLfarm* のローカルな各実サーバの同じ IP アドレスまたはポートを指定して、仮想サーバを作成する必要があります。この仮想サーバは内部で、その仮想サーバ宛てのあらゆるトラフィックを SSLSM に転送するように設定されます。

内部生成の形でコンフィギュレーションを作成しなければならないのは、ローカル実サーバの IP アドレスと CSM 仮想サーバのアドレスを一致させなければならないからです。CSM がこのローカル実サーバへの接続を開始すると、CSM が SYN（同期）フレームを送受信します。CSM が SYN を受信し、宛先 IP アドレスまたはポートが仮想サーバの VS1 と同じだった場合、CSM はさらに具体的な仮想サーバが追加されないかぎり、VS1 と一致したとみなします。

サーバ側の設定

SSLSM がバックエンド サーバとして CSM を使用する場合、レイヤ 4 およびレイヤ 7 のロードバランシングには、仮想サーバの標準設定を使用します。

次に、SSLSM からのトラフィックだけを受信するように、この仮想サーバに制限を加える例を示します。

```
Router(config-module-csm)# serverfarm SLBdefaultfarm
Router(config-slb-sfarm)# real 10.2.0.20
Router(config-slb-sfarm)# inservice

Router(config-module-csm)# vserver VS2
Router(config-slb-vserver)# virtual 10.2.0.100 tcp www
Router(config-slb-vserver)# serverfarm SLBdefaultfarm
Router(config-slb-vserver)# vlan local
Router(config-slb-vserver)# inservice
```

次に、バックエンド サーバとして実サーバを設定する例を示します。

```
Router(config-module-csm)# serverfarm SSLpredictorforward
Router(config-slb-sfarm)# predictor forward

Router(config-module-csm)# vserver VS3
Router(config-slb-vserver)# virtual 0.0.0.0 0.0.0.0 tcp www
Router(config-slb-vserver)# serverfarm SSLpredictorforward
Router(config-slb-vserver)# inservice
```

バックエンド サーバとしての CSM の設定

仮想サーバおよびサーバ ファームの設定により、実サーバをバックエンド サーバとして使用できます。CSM をバックエンド サーバとして使用するには、「クライアント側の設定」（p.8-35）で説明した設定を使用し、さらに SSL daughter card を設定します。

次に、CSM 仮想サーバにレイヤ 7 ロードバランシングを設定する例を示します。

```
Router(config-module-csm)# serverfarm SLBdefaultfarm
Router(config-slb-sfarm)# real 10.2.0.20
Router(config-slb-real)# inservice

Router(config-module-csm)# serverfarm SLBjpgfarm
Router(config-slb-sfarm)# real 10.2.0.21

Router(config-module-csm)# map JPG url
Router(config-slb-map-cookie)# match protocol http url *jpg*

Router(config-module-csm)# policy SLBjpg
Router(config-slb-policy)# url-map JPG
Router(config-slb-policy)#serverfarm SLBjpgfarm

Router(config-module-csm)# vserver VS2
Router(config-slb-vserver)# virtual 10.2.0.100 tcp www
Router(config-slb-vserver)# serverfarm SLBdefaultfarm
Router(config-slb-vserver)# slb-policy SLBjpg
Router(config-slb-vserver)# inservice
```

次に、CSM 仮想サーバにレイヤ 4 ロードバランシングを設定する例を示します。

```
Router(config-module-csm)# serverfarm SLBdefaultfarm
Router(config-slb-sfarm)# real 10.2.0.20
Router(config-slb-real)# inservice

Router(config-module-csm)# vserver VS2
Router(config-slb-vserver)# virtual 10.2.0.100 tcp www
Router(config-slb-vserver)# serverfarm SLBdefaultfarm
Router(config-slb-vserver)# vlan local
Router(config-slb-vserver)# inservice
```

バックエンド サーバとしての実サーバの設定

実サーバをバックエンド サーバとした、サーバ側コンフィギュレーションのトラフィック フローは、クライアント側のコンフィギュレーションと同様です。実サーバをバックエンド サーバとして使用するには、「クライアント側の設定」（p.8-35）で説明した設定を使用し、さらに SSLSM を設定します。

SSLSM プロキシ サービスに関して、新しい設定は不要です。次に、設定を内部で開始し、ユーザにわからないようにする例を示します。

```
ssl-proxy(config)# ssl-proxy service S1
ssl-proxy(config-ssl-proxy)# virtual ipaddr 10.1.0.21 protocol tcp port 443 secondary
ssl-proxy(config-ssl-proxy)# server ipaddr 10.2.0.20 protocol TCP port 80
ssl-proxy(config-ssl-proxy)# inservice
```

次に、CSM 仮想サーバの設定例を示します。

```
Router(config-module-csm)# serverfarm SSLreals

Router(config-slb-sfarm)# real 10.2.0.20
Router(config-slb-sfarm)# inservice

Router(config-module-csm)# serverfarm SSLpredictorforward
Router(config-slb-sfarm)# predictor forward

Router(config-module-csm)# vserver VS3
Router(config-slb-vserver)# virtual 0.0.0.0 0.0.0.0 tcp www
Router(config-slb-vserver)# serverfarm SSLpredictorforward
Router(config-slb-vserver)# inservice
```

ヘルス モニタリングの設定

この章では、Content Switching Module（CSM; コンテント スイッチング モジュール）上でヘルス モニタリングを設定する方法について説明します。

ヘルス モニタリング用プローブの設定

実サーバへのヘルス プローブを設定すると、実サーバが正しく動作しているかどうかを調べることができます。実サーバの状態は次のように分類されます。

- アクティブ ― 実サーバが適切に応答します。
- サスペクト ― 実サーバに到達できないか、または無効な応答が戻ります。プローブは再試行されます。
- 失敗 ― 指定回数だけ続けて再試行したあと、実サーバは応答に失敗します。失敗という情報が通知されると、CSM は着信接続を相応に調整します。サーバが再びアクティブになるまで、プローブは失敗状態のままです。

CSM は、実サーバのモニタに使用されるプローブをサポートします。プローブを設定する手順は、次のとおりです。

- プローブ サブモードを開始します。
- プローブに名前を付けます。
- プローブのタイプを指定します。

CSM は FTP、Domain Name System（DNS; ドメイン ネーム システム）、HTTP など、実サーバをモニタするさまざまなプローブ タイプをサポートします。

（注） デフォルトでは、CSM 上にプローブは設定されていません。

ヘルス プローブ モニタリング用に CSM を設定する場合は、次の動作を含む多層的アプローチを使用することができます。

- アクティブ プローブ ― これらのプローブを定期的に実行します。Internet Control Message Protocol（ICMP）、TCP、HTTP、およびその他の前もって定義されたヘルス プローブはこのカテゴリに入ります。スクリプト化されたヘルス プローブもここに入ります。アクティブなプローブは、セッションのセットアップまたはシステムのティアダウンに影響しません。
- パッシブ モニタリング（帯域内ヘルス モニタリング）― サーバをサービス対象外にする可能性のある破壊的なエラー（たとえば、サーバからのリセット［RST］やサーバからの応答なし）が発生していないかセッションをモニタします。これらのヘルス チェックは、フルセッション レートで動作し、サーバの故障を迅速に認識します。
- パッシブ HTTP エラー コード チェック（帯域内応答解析）― CSM は、HTTP 戻りコードを解析し、サーバが使用不可になる「サービス利用不可」のようなコードを監視します。パッシブ HTTP エラー コード チェックは、セッションのパフォーマンスにはほとんど影響しません。

プローブを用意するには、プローブ サブモードでプローブに名前を付け、プローブ タイプを指定して、プローブを設定する必要があります。

プローブを設定したあとで、プローブを実サーバ ファームに関連付けて、有効にする必要があります。サーバ ファーム内のすべてのサーバは、このサーバ ファームに関連付けられたプローブ タイプのプローブを受信します。1 つのサーバ ファームに 1 つまたは複数のプローブ タイプを関連付けることができます。

実サーバまたは仮想サーバの設定時にポート番号を割り当てる場合は、プローブの設定時にポート番号を指定する必要はありません。プローブは実サーバまたは仮想サーバ コンフィギュレーションからポート番号を引き継ぎます。

ヘルス プローブ コンフィギュレーションでオプションのヘルス プローブ ポート機能を使用し、プローブにポートを明示的に指定することによって、実サーバまたは仮想サーバのポート情報を変更できます。この機能を使用すると、実サーバまたは仮想サーバでポートが指定されていない場合に、ヘルス プローブに使用させるポートを設定できます。

プローブを設定してから、サーバ ファームに 1 つまたは複数のプローブを関連付けます。サーバ ファーム内のすべてのサーバは、このプールに関連付けられたプローブ タイプのプローブを受信します。

（注） 対応するサービスが動作していない実サーバを含むサーバ ファームに特定タイプのプローブが関連付けられている場合に、そのタイプのプローブを受信すると、実サーバはエラー メッセージを送信します。その結果、CSM が実サーバを失敗ステートにして、サーバ ファームからその実サーバを使用できないようにします。

プローブのタイプおよび名前を指定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# probe` *probe-name* {**http** | **icmp** | **telnet** | **tcp** | **ftp** | **smtp** | **dns** | **kal-ap-upd**}	プローブのタイプおよび名前を指定します[1]、[2]。 • *probe-name* は、設定中のプローブの名前を表す最大 15 文字の文字列です。 • **http** を指定すると、デフォルト設定の HTTP プローブが作成されます。 • **icmp** を指定すると、デフォルト設定の ICMP プローブが作成されます。 • **telnet** を指定すると、デフォルト設定の Telnet プローブが作成されます。 • **tcp** を指定すると、デフォルト設定の TCP プローブが作成されます。 • **ftp** を指定すると、デフォルト設定の FTP プローブが作成されます。 • **smtp** を指定すると、デフォルト設定の SMTP プローブが作成されます。 • **dns** を指定すると、デフォルト設定の DNS プローブが作成されます。 • **kal-ap-upd** を指定すると、Global Server Load Balancing（GSLB; グローバル サーバ ロードバランシング）ターゲット プローブが作成されます。
ステップ 2	`Router(config-slb-probe-tcp)# port` *port-number: 1-MAXUSHORT*	プローブ用のオプション ポートを設定します[3]。
ステップ 3	`Router# show module csm` *slot* `probe`	すべてのプローブおよびその設定を表示します。
ステップ 4	`Router# show module csm` *slot* `tech-support probe`	プローブの統計情報を表示します。

1. 設定からプローブ タイプを削除するには、**no** 形式を使用します。
2. パフォーマンス アラートを受信した場合、帯域内ヘルス モニタリングの方が、よりスケーラブルなソリューションになります。
3. **port** コマンドは、ICMP または PING ヘルス プローブには使用できません。

（注） プローブの名前およびタイプを指定すると、最初にデフォルト値が設定されます。デフォルト設定を変更するには、プローブ コンフィギュレーション コマンドを入力します。

次に、プローブを設定する例を示します。

```
Router(config-module-csm)# probe probe1 tcp
Router(config-slb-probe-tcp)# interval 120
Router(config-slb-probe-tcp)# retries 3
Router(config-slb-probe-tcp)# failed 300
Router(config-slb-probe-tcp)# open 10
Router(config-slb-probe-tcp)# serverfarm sf4
Router(config-slb-sfarm)# real 10.1.0.105
Router(config-slb-real)# inservice
Router(config-slb-real)# probe probe1
Router(config-slb-sfarm)# vserver vs4
Router(config-slb-vserver)# virtual 10.1.0.84 tcp 80
Router(config-slb-vserver)# serverfarm sf4
Router(config-slb-vserver)# inservice
Router(config-slb-vserver)# end
```

（注） タイムアウト値には open と receive の 2 種類があります。open のタイムアウトでは、接続のオープンを待機する秒数（SYN の送信後、SYN ACK を待つ秒数）を指定します。receive のタイムアウトでは、データの受信を待機する秒数（GET/HEAD 要求の送信後、HTTP 応答を待つ秒数）を指定します。TCP プローブは、オープンするとデータを送信しないでただちにクローズされるので、receive のタイムアウトは使用しません。

プローブ コンフィギュレーション コマンド

次に示すコマンドはすべてのプローブ タイプに共通です。

コマンド	目的
Router(config-slb-probe)# **interval** *seconds*	プローブとプローブの間隔（前のプローブの終了から次のプローブの開始までの期間）を秒単位で設定します[1]、[2]。 範囲 = 2 ～ 65535 秒 デフォルト = 120 秒
Router(config-slb-probe)# **retries** *retry-count*	サーバを失敗としてマークするまでに許容されたプローブの失敗回数を設定します[1]。 範囲 = 0 ～ 65535 デフォルト = 3
Router(config-slb-probe)# **failed** *failed-interval*	サーバが失敗としてマークされた場合にヘルス チェックを行う間隔を設定します。この時間は秒単位です[1]。 範囲 = 2 ～ 65535 デフォルト = 300 秒
Router(config-slb-probe)# **recover** *recover_value*	失敗状態のサーバを正常としてマークするまでに連続して受信する応答数を設定します[1]。 範囲 = 1 ～ 65535 デフォルト = 1
Router(config-slb-probe)# **open** *open-timeout*	TCP 接続を待機する最大時間を設定します。このコマンドは TCP 以外のヘルス チェック（ICMP または DNS[1]）には使用しません。 範囲 = 1 ～ 65535 デフォルト = 10 秒
Router(config-slb-probe)# **description** *description*	（任意）プローブの説明を指定します。*description* は最大 80 文字までです。

1. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。
2. パフォーマンス アラートを受信した場合、帯域内ヘルス モニタリングの方が、よりスケーラブルなソリューションになります。

HTTP プローブの設定

HTTP プローブは実サーバに対する HTTP 接続を確立し、HTTP 要求を送信して、応答を確認します。**probe** *probe-name* **http** コマンドを実行すると、HTTP プローブ コンフィギュレーション サブモードが開始されます。

HTTP プローブを設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# probe probe-name http`	HTTP プローブを設定して、HTTP プローブ サブモードを開始します[1]。
ステップ 2	`Router(config-slb-probe-http)# credentials username [password]`	HTTP SLB プローブの基本認証値を設定します[1]。
ステップ 3	`Router(config-slb-probe-http)# expect status min-number [max-number]`	HTTP プローブから戻ることが予測されるステータス コードを設定します。**expect status** コマンドを一度に 1 つずつ入力することによって、複数のステータス範囲を設定できます[1]。 *min-number* ― *max-number* を指定しなかった場合、この値が単一のステータス コードになります。最大数を指定した場合、この値がステータス コード範囲の下限になります。 *max-number* ― ステータス コード範囲の上限です。デフォルト値は 0 ～ 999（サーバからの応答はどれも有効）です。 **（注）** 上限を指定しなかった場合、このコマンドには単一の値（min-number）が設定されます。最小数と最大数の両方を指定した場合、このコマンドには範囲が設定されます。
ステップ 4	`Router(config-slb-probe-http)# header field-name [field-value]`	HTTP プローブのヘッダー フィールドを設定します。複数のヘッダー フィールドを指定できます[1]。
ステップ 5	`Router(config-slb-probe-http)# request [method [get|head]] [url path]`	HTTP プローブで使用する要求メソッドを設定します[1]。 • **get** ― HTTP **get** 要求メソッドは、サーバにこのページの取得を指示します。 • **head** ― HTTP **head** 要求メソッドは、このページのヘッダーだけを取得するように、サーバに指示します。 • **url** ― 最大 1275 文字の文字列で URL パスを指定します。デフォルトでは「"/"」です。 **（注）** CSM がサポートするのは、**get** および **head** 要求メソッドだけです。**post** およびその他のメソッドはサポートしません。デフォルトのメソッドは **get** です。

1. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。

ICMP プローブの設定

ICMP プローブは ICMP エコー（ping など）を実サーバに送信します。**probe icmp** コマンドを入力すると、ICMP プローブ コンフィギュレーション モードが開始されます。一般的な **probe** コマンドはすべてサポートされていますが、**open** コマンドはサポートされずに無視されます。

UDP プローブを設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# probe` *probe-name* **icmp**	ICMP プローブを設定して、ICMP プローブ サブモードを開始します[1]。
ステップ 2	`Router(config-slb-probe-icmp)# interval`	失敗したサーバのプローブ間、およびプローブ間の待機間隔を設定します。
ステップ 3	`Router(config-slb-probe-icmp)# receive`	TCP 接続にサーバからの応答を受信させる時間を指定します。
ステップ 4	`Router(config-slb-probe-icmp)# retries`	サーバが失敗とみなされるまでの再試行回数を制限します。

1. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。

UDP プローブの設定

ICMP がないと UDP プローブはサーバのダウンまたはサーバの切断を検出できないため、UDP プローブには ICMP が必要です。UDP をスーパバイザ エンジンに関連付けて、ICMP を設定する必要があります。

UDP プローブは Raw UDP プローブなので、CSM はプローブの応答ペイロードに単一のバイトを使用します。CSM は UDP アプリケーションから意味のある応答がくることを想定していません。CSM は ICMP 到達不能メッセージを使用して、UDP アプリケーションが到達可能かどうかを判断します。受信タイムアウトで ICMP 到達不能の応答がない場合、CSM はプローブが正常に実行されていると判断します。実サーバの IP インターフェイスがダウンまたは切断された場合、UDP プローブは自身で UDP アプリケーションが到達不能にあることを判断できません。指定のサーバの UDP プローブのほかに ICMP プローブを設定する必要があります。

CSM は高レベル UDP アプリケーションとして DNS プローブを使用します。Toolkit Command Language（TCL）スクリプトを使用すると、このプローブを設定できます。第 10 章「CSM での TCL スクリプトの使用」を参照してください。

UDP プローブを設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# probe` *probe-name* **udp**	UDP プローブを設定して、UDP プローブ サブモードを開始します[1]。
ステップ 2	`Router(config-slb-probe-icmp)# interval`	失敗したサーバのプローブ間、およびプローブ間の待機間隔を設定します。
ステップ 3	`Router(config-slb-probe-icmp)# receive`	TCP 接続にサーバからの応答を受信させる時間を指定します。
ステップ 4	`Router(config-slb-probe-icmp)# retries`	サーバが失敗とみなされるまでの再試行回数を制限します。

1. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。

TCP プローブの設定

TCP プローブは接続を確立および解除します。**probe tcp** コマンドを入力すると、TCP プローブ コンフィギュレーション モードが開始されます。一般的な **probe** コマンドはすべてサポートされます。

TCP プローブを設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# probe` *probe-name* **tcp**	TCP プローブを設定して、TCP プローブ サブモードを開始します[1]。
ステップ 2	`Router(config-slb-probe-icmp)# interval`	失敗したサーバのプローブ間、およびプローブ間の待機間隔を設定します。
ステップ 3	`Router(config-slb-probe-icmp)# retries`	サーバが失敗とみなされるまでの再試行回数を制限します。

1. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。

FTP、SMTP、および Telnet プローブの設定

FTP、SMTP、または Telnet プローブは実サーバへの接続を確立し、アプリケーションからグリーティングが届いたかどうかを検証します。**probe**（**ftp**、**smtp**、または **telnet**）コマンドを実行すると、対応するプローブ コンフィギュレーション モードが開始されます。一般的な **probe** コマンド オプションはすべてサポートされます。一度に 1 つずつコマンドを実行することによって、複数のステータス範囲を指定できます。

FTP、SMTP、または Telnet プローブから戻ることが予測されるステータス コードを設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# probe` *probe-name* [**ftp** | **smtp** | **telnet**]	FTP、SMTP、または Telnet プローブを設定し、FTP、SMTP、または Telnet プローブ サブモードを開始します[1]。
ステップ 2	`Router(config-slb-probe-icmp)# interval`	失敗したサーバのプローブ間、およびプローブ間の待機間隔を設定します。
ステップ 3	`Router(config-slb-probe-icmp)# receive`	TCP 接続にサーバからの応答を受信させる時間を指定します。
ステップ 4	`Router(config-slb-probe-icmp)# retries`	サーバが失敗とみなされるまでの再試行回数を制限します。

1. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。

DNS 解決要求の指定

DNS プローブは実サーバにドメイン名解決要求を送信し、戻された IP アドレスを確認します。**probe dns** コマンドを実行すると、DNS プローブ コンフィギュレーション サブモードが開始されます。一般的な probe コマンドはすべてサポートされますが、**open** はサポートされずに無視されます。

ドメイン名解決要求を指定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Router(config-module-csm)# probe` *probe-name* **dns**	DNS プローブを設定して、DNS プローブ サブモードを開始します[1]。
ステップ 2	`Router(config-slb-probe-dns)#` [**failed** | **interval** | **retries** | **receive**]	DNS 接続を行うためのプローブ間の待機間隔、サーバから応答を受信する時間、および実サーバが失敗したとみなされるまでの再試行回数の上限を指定します。

1. デフォルトの設定に戻すには、このコマンドの **no** 形式を使用します。

帯域内ヘルス モニタリングの設定

ここでは、帯域内ヘルス モニタリングについて説明します。

帯域内ヘルス モニタリングの概要

接続のバランスを図るために、CSM はコンフィギュレーションに含まれているすべての実サーバの状態をたえずモニタしていなければなりません。帯域内ヘルス モニタリング機能を各サーバ ファームに設定し、サーバの状態をモニタします。サーバ ファームごとに設定されたパラメータは、そのサーバ ファーム内の個々の実サーバに適用されます。システムが実サーバを到達不能とみなすまでのセッションの異常終了の回数を設定できます。また、実サーバがサーバ ファームに再度導入され、接続が試行されるまでの待機時間を指定することもできます。

この機能は、ヘルス プローブと連動します。あるサーバにヘルス プローブと帯域内ヘルス モニタリングを両方とも設定した場合、サーバ ファーム内の実サーバでサービスを維持するには、両方のヘルス チェック セットを実行する必要があります。どちらか一方のヘルス チェック機能でサーバの停止が検出された場合、CSM はロードバランスの対象としてそのサーバを選択しません。

帯域内ヘルス モニタリングの設定

帯域内ヘルス モニタリングを設定する手順は、次のとおりです。

ステップ 1　サーバ ファームが設定されていることを確認します（「サーバ ファームの設定」[p.5-2] を参照）。

ステップ 2　**serverfarm** サブモード コマンドを入力し、各サーバ ファームに対して帯域内ヘルス モニタリングをイネーブルにします。

```
Router(config-module-csm)# serverfarm serverfarm-name
Router(config-slb-sfarm)# health retries count failed seconds
```

（注）　retries では、実サーバをサービスから除外するまでに CSM が許容する、セッション異常終了の回数を指定します。failed では、帯域内ヘルス チェックによってサービスから除外された実サーバへの接続を再試行するまでに、CSM が待機する時間を秒単位で指定します。

次に、geo というサーバ ファームに対して帯域内ヘルス モニタリングをイネーブルにする例を示します。

```
Router(config-module-csm)# serverfarm geo
Router(config-slb-sfarm)# health retries 43 failed 160
```

HTTP 戻りコード チェックの設定

ここでは、HTTP 戻りコードのチェック機能について説明します。

HTTP 戻りコード チェックの概要

戻りエラー コード チェック（戻りコード解析）機能は、サーバがいつ Web ページを正しく戻さなかったかを調べる場合に使用します。この機能は、CSM のパケットを調べる機能を拡張し、HTML 戻りコードを解析し、サーバが戻した戻りコードに作用します。

CSM から HTTP 要求を受信したサーバは、HTTP 戻りコードを使用して応答します。CSM はこの HTTP 戻りエラー コードを使用することによって、サーバの可用性を判別できます。特定の戻りコードを受け取った場合に、サーバを使用停止にするように CSM を設定できます。

RFC 2616 で、定義済みコードのリスト（100 ～ 599）が指定されています。戻りコードのチェックでは、ほかより有用なコードがいくつかあります。たとえば、404 という戻りコードは、URL が見つからなかったという定義です。これは、ユーザが URL を正しく入力しなかった結果である可能性があります。エラー コード 404 は、不良ディスク ドライブが原因で Web サーバが要求されたデータを見つけられなかった場合など、Web サーバのハードウェアの問題を意味することもあります。この場合、Web サーバそのものはアクティブですが、ディスク ドライブが故障しているので、要求されたデータを送信できません。サーバがデータを戻せないかぎり、そのサーバには今後、データ送信を要求しないでおくべきです。戻りコード チェックで使用するエラー コードを特定する場合は、RFC 2616 を参照してください。

HTTP 戻りコード チェックを設定すると、CSM はロードバランス対象のすべての HTTP 接続から送られた HTTP 応答をモニタし、実サーバごとに戻りコードの発生を記録します。CSM は戻りコードのカウントを保存します。戻りコードがスレッシュホールドに達した場合、CSM は Syslog メッセージを送信したり、またはサーバをサービスから除外したりすることができます。

デフォルトのアクション、戻りコードのカウント、Syslog メッセージを適用できます。また、実サーバをサービスから除外することもできます。サーバ ファームには、これらのアクションのいずれか、またはこれらのアクションのセットを適用できます。1 つの仮想グループを複数のサーバ ファームにバインドすることによって、複数のサーバ ファームで 1 つの戻りコード サーバ ファーム ポリシーを再利用することもできます。

（注） 仮想サーバに HTTP 戻りコード チェックを設定すると、その仮想サーバのパフォーマンスが低下します。戻りコードの解析をイネーブルにした場合、あらゆる HTTP サーバ応答で戻りコードを解析しなければなりません。

HTTP 戻りコード チェックの設定

戻りエラー コード チェックを設定するには、サーバ ファームの属性を設定し、サーバ ファームを戻りコード マップと関連付ける必要があります。

戻りコード チェックを設定する手順は、次のとおりです。

ステップ 1　HTTP 仮想サーバが設定されていることを確認します（「リダイレクト仮想サーバの設定」[p.6-8] を参照）。

ステップ 2　マップ戻りコード コマンドを入力して戻りコードのマッピングをイネーブルに設定し、戻りコード マップ サブモードを開始します。

```
Router(config-module-csm)# map name retcode
```

ステップ 3　戻りコードの解析を設定します。

```
Router(config-slb-map-retcode)# match protocol http retcode min max action [count |
log | remove] threshold [reset seconds]
```

マップで必要とされる一致数を設定できます。

ステップ 4　戻りコード マップをサーバ ファームに割り当てます。

```
Router(config-slb-sfarm)# retcode-map name
```

次に、戻りエラー コード チェックをイネーブルにする例を示します。

```
Router(config-module-csm)# map httpcodes retcode
Route(config-slb-map-retcode)# match protocol http retcode 401 401 action log 5 reset
120
Route(config-slb-map-retcode)# match protocol http retcode 402 415 action count
Route(config-slb-map-retcode)# match protocol http retcode 500 500 action remove 3
reset 0
Route(config-slb-map-retcode)# match protocol http retcode 503 503 action remove 3
reset 0
Route(config-slb-map-retcode)# exit
Router(config-module-csm)# serverfarm farm1
Router(config-slb-sfarm)# retcode-map httpcodes
Router(config-slb-sfarm)# exit
Router(config-module-csm)# end
```

CHAPTER 10

CSM での TCL スクリプトの使用

この章では、コンテント スイッチングの設定方法について説明します。

Content Switching Module（CSM; コンテント スイッチング モジュール）により、Toolkit Command Language（TCL）スクリプトをアップロードし、CSM 上で実行できます。TCL は、ネットワーク コミュニティで普及しているスクリプト言語です。また、TCL には開発されたスクリプトに関連する大量のライブラリがあり、さまざまなサイトから容易にアクセスできます。TCL スクリプトを使用すると、カスタム TCL スクリプトを作成し、カスタム ヘルス プローブまたはスタンドアロン タスクを開発することができます。

CSM の TCL インタープリタ コードは、標準 TCL の Release 8.0 に準拠しています。スクリプトを作成してヘルス プローブを設定するか（「ヘルス モニタリング用プローブの設定」[p.9-2] を参照）、またはヘルス プローブに含まれないタスクを CSM 上で実行することができます。CSM はユーザ側で設定可能な間隔で、定期的にスクリプトを実行します。

CSM Release 3.1(1a) までは、基本ヘルス モニタリング コードに含まれないプロトコル用にヘルス プローブを設定することはできませんでした。現在は、プローブを作成し、特定のアプリケーションに合わせて CSM をカスタマイズできます。CSM Release 3.2 は、UDP ソケット機能をサポートします。

CSM は現在、次の 2 種類のスクリプト モードをサポートします。

- プローブ スクリプト モード — このタイプのスクリプトは、ある種の単純なルールに従って作成する必要があります。このスクリプトの実行は、ヘルス モニタリング モジュールが制御します。

  スクリプトは、スクリプト プローブの一部として定期的に実行され、実行中のスクリプトが返した終了コードによって、特定の実サーバについて、相対的な状態と可用性がわかります。スクリプト プローブの動作は、現在の CSM ソフトウェアで利用できる他のヘルス プローブと同様です。
- スタンドアロン スクリプト モード — このタイプのスクリプトは、一般的な TCL スクリプトです。このスクリプトの実行は、CSM のコンフィギュレーションによって制御します。プローブ スクリプトはスタンドアロン タスクとして実行できます。

TCL 機能をサポートするために、サンプル スクリプトを利用できます。その他のカスタム スクリプトも使用できますが、これらのサンプル スクリプトはシスコシステムズの TAC がサポートしています。サンプル スクリプトのファイルには、次の URL からアクセスしてください。

http://www.cisco.com/cgi-bin/tablebuild.pl/cat6000-intellother

スクリプト ファイルの名前は、c6slb-script.3-3-1.tcl です。

スクリプトのロード

スクリプトは、スクリプト ファイルによって CSM にロードします。スクリプト ファイルには、スクリプトがない場合もあれば、1 つまたは複数のスクリプトが含まれている場合もあります。1 つのスクリプトに 128 KB のスタック スペースが必要です。ヘルス スクリプトは最大 50 ありますので、スクリプト プローブのスタックスペースは、最大で 6.4 MB になります。スタンドアロン スクリプトも実行できますが、そのためにさらにスタック スペースを消費します。

スクリプトのロード例

スクリプトは TFTP サーバ、ブートフラッシュ、スロット 0、またはその他のストレージ デバイスから、**script file** [*file-url*] コマンドを使用してロードできます。

次に、スクリプトをロードする例を示します。

```
Router(config)# module csm 4
Router(config-module-csm)# script file tftp://192.168.1.1/httpProbe.test
```

スクリプト名は、スクリプトのファイル名またはスクリプト ファイル内部で符号化された特殊な名前です。各スクリプト ファイルは、同じファイルに複数のスクリプトが収められている場合があります。スクリプトを実行する、またはスクリプトを使用してヘルス プローブを作成するには、スクリプトのロード元となったスクリプトファイルではなく、スクリプト名を参照する必要があります。

各関連スクリプトを識別できるように、次の行から各スクリプトを始める必要があります。

```
#!name = script_name
```

スクリプトをバンドルしたマスター スクリプト ファイルの例を示します。

```
#!name = SCRIPT1
puts "this is script1"
!name = SCRIPT2
puts "this is script2"
```

マスター スクリプト ファイルに含まれているファイルを調べる例を示します。

```
Router(config)# configure terminal
Router(config-t)# module csm 4
Router(config-module-csm)# script file tftp://192.168.1.1/script.master
Router(config-module-csm)# end
```

次の例では、script.master ファイルにスクリプトが 3 つあります。

```
Router(config)# show module csm 4 file tftp://192.168.1.1/script.master
script1, file tftp://192.168.1.1/script.master
  size = 40, load time = 03:49:36 UTC 03/26/93
script2, file tftp://192.168.1.1/script.master
  size = 40, load time = 03:49:36 UTC 03/26/93
```

ロードしたスクリプト ファイルの内容を表示するには、次のコマンドを使用します。

```
Router(config)# show module csm slot script full_file_URL code
```

指定のスクリプト内のコードを表示する例を示します。

```
router1# show module csm 6 script name script1 code
script1, file tftp://192.168.1.1/script.master
  size = 40, load time = 03:04:36 UTC 03/06/93
#!name = script1
```

スタンドアロン スクリプト タスクとスクリプト プローブの大きな相違は、ヘルス モニタリング用の CSM モジュールによって、ヘルス スクリプトがスケジューリングされるかどうかです。次の条件が当てはまります。

- スクリプト プローブがアクティブでも、スクリプトを変更できます。変更点は、次のスクリプト実行時に、コマンド ライン引数に自動的に適用されます。
- プローブの設定時に、特定のスクリプトがプローブに結合されます。その時点でスクリプトが使用できないと、プローブは NULL スクリプトを使用して実行されます。この状況が発生すると、警告フラグが生成されます。ただし、スクリプトのリロード時には、プローブ オブジェクトとスクリプト間のバインディングは自動的には実行されません。バインディングを実行するには、もう一度、**no script** および **script** コマンドを使用する必要があります。
- スクリプトがロードされたあとは、システムに残り、削除できません。スクリプトを変更するには、スクリプトを変更してから再び **no script file** および **script file** コマンドを入力します。
- 各スクリプトは常に固有の名前で識別されます。同名のスクリプトが 2 つ以上ある場合、CSM は最後にロードされたスクリプトを使用します。スクリプト名が重複していると、CSM によって警告メッセージが生成されます。

TCL スクリプトのリロード

スクリプト ファイルをロードすると、スクリプトのロード元ファイルとは無関係に、そのファイルに含まれていたスクリプトが CSM に組み込まれます。その後、スクリプト ファイルを変更する場合には、**script file** コマンドを使用して、スクリプト ファイルをリロードし、CSM 上で変更できるようにします（詳細については、『*Catalyst 6500 Series Content Switching Module Command Reference*』を参照してください）。次に、スクリプトをリロードする例を示します。

```
router(config)# module csm 4
router(config-module-csm)# no script file tftp://192.168.1.1/script.master
router(config-module-csm)# script file tftp://192.168.1.1/script.master
Loading script.master from 192.168.1.1 (via Vlan100): !!!!!!!!!!!!!!!
[OK - 74804 bytes]
router(config-module-csm)# end
```

実行コンフィギュレーションから **script file** コマンドを削除する場合は、**no script file** コマンドを使用します。このコマンドによってファイルに含まれていたスクリプトがアンロードされるわけではありません。また、CSM 上で実行中のスクリプトも影響を受けません。ロードしたスクリプトをアンロードすることはできません。ロードしたスクリプトが不要になっても、スクリプトを削除する必要はありません。

TCL スクリプトおよび CSM

CSM Release 4.1(1) の TCL スクリプト機能は、TCL 8.0 ソース ディストリビューション ソフトウェアに準拠しています。CSM TCL の変更により、スタンダード TCL ライブラリとは異なる他のプロセスを呼び出すために中断したり、同時に TCL インタープリタを実行したりすることが可能になりました。CSM TCL ライブラリは、**file**、**fcopy**、およびその他の標準 TCL ファイル入出力コマンドをサポートしません。

表 10-1 に、CSM がサポートする TCL コマンドを示します。

表 10-1　CSM がサポートする TCL コマンド

コマンド			
一般的な TCL コマンド			
append	array	binary	break
catch	concat	continue	error
eval	exit	expr	fblocked
for	foreach	format	global
gets	if	incr	info
join	lappend	lindex	linsert
list	llength	lrange	lreplace
lsearch	lsort	proc	puts
regexp	regsub	rename	return
set	split	string	subst
switch	unset	uplevel	upvar
variable	while	namespace	
時間関連のコマンド			
after	clock	time	
ソケット コマンド			
close	blocked	fconfigured	fileevent
flush	eof	read	socket
update	vwait		

表 10-2 に、CSM がサポートしない TCL コマンドを示します。

表 10-2　CSM がサポートしない TCL コマンド

一般的な TCL コマンド			
cd	fcopy	file	open
seek	source	tell	filename
load	package		

表 10-3 に、CSM 固有の TCL コマンドを示します。

UDP コマンド セットにより、Scotty ベースの TCL スクリプトが CSM 上で実行されます。Scotty はソフトウェア パッケージの名前で、高レベル、ストリングベースの API を使用するサイト特有のネットワーク管理ソフトウェアの実装を可能にします。すべての UDP コマンドは、その他の CSM TCL コマンド同様、スレッド セーフ（複数のプログラム間でデータ共有が可能）です。

表 10-3　CSM 固有の TCL コマンド

コマンド	定義
disable_real *serverfarmName realIp port ,-1* | **all** *probeNumId probeNameId*	PROBE_FAIL ステートにすることで、サーバ ファームの実サーバをディセーブルにします。成功した場合、このコマンドは 1 を返します。失敗した場合は 0 を返します。 `disable_real SF_TEST 1.1.1.1 -1 10 cisco` **（注）** サーバ ファーム名は、CSCec72471 ごとに大文字を使用する必要があります。
enable_real *serverfarmName realIp port ,-1* | **all** *probeNumId probeNameId*	PROBE_FAIL ステートから動作可能な状態に実サーバをイネーブルにします。成功した場合、このコマンドは 1 を返します。失敗した場合は 0 を返します。 `enable_real SF_TEST 1.1.1.1 -1 10 cisco` **（注）** サーバ ファーム名は、CSCec72471 ごとに大文字を使用する必要があります。
gset *varname value*	同じスクリプトで実行されているすべてのプローブ スレッドに対してグローバルな変数を設定することで、プローブ状態を保存できます。このコマンドは、プローブ スクリプトに対してのみ機能します。スタンドアロン スクリプトには適用されません。 プローブ スクリプトの変数は、1 つのプローブ スレッド内でしか認識できません。プローブが終了するたびに、すべての変数は消去されます。たとえば、プローブ スクリプトに「gset x 1 ; incr x」が含まれている場合、変数 x にはプローブごとに 1 が追加されます。 • スクリプトから変数の値を取得するには、*var* または $*var* を設定します。 • スクリプトから変数の値をリセットするには、unset *var* を設定します。 • 現在の変数の値を表示するには、**show module csm** *slot* **tech script** コマンドを使用します。詳細については、「プローブ スクリプトのデバッグ」（p.10-13）を参照してください。

表 10-3　CSM 固有の TCL コマンド（続き）

コマンド	定義
socket -graceful host A.B.C.D port	デフォルトでは、すべての CSM スクリプト プローブがリセットの送信によって TCP ソケットをクローズしています。このアクションは、CSM がアクティブ TCP クローズを初期化するときに、TIME_WAIT ステートを回避するために実行されます。 VxWork で使用できるソケット数が 255 に制限されているため、多くのプローブが同時に実行されている場合、CSM のシステム リソースが足りなくなり、ソケットのオープン時に次のプローブを実行できなくなります。 ソケットに適切なコマンドが入力されると、CSM はリセットの代わりに FIN を使用して TCP 接続をクローズします。このコマンドは、システムのプローブが 250 未満の場合のみ使用します。 `set sock [socket -graceful 192.168.1.1 23]`
ping [*numpacket*] *host A.B.C.D*	このコマンドは CSM Release 3.2 では現在使用できません。 スクリプトのホストに ping を実行できます。成功した場合、このコマンドは 1 を返します。失敗した場合は 0 を返します。 `set result [ping 3 1.1.1.1]` **（注）** リモート ホストが各 CSCea67098 の CSM と同じサブネットにない場合、このコマンドはスクリプトをブロックします。

xml *xmlConfigString*	TCL スクリプトから CSM に XML 設定ストリングを送信します。このコマンドは、XML サーバが CSM でイネーブルになっている場合のみ機能します。XML 設定の章を参照してください。
	
	このコマンドは、XML 設定の結果のストリングを返します。
	
	`set cfg_result [ xml {`
	`<config>`
	`<csm_module slot="6">`
	`<serverfarm name="SF_TEST">`
	`</serverfarm>`
	`</config>`
	`}`
	`]`

表 10-4 に、CSM で使用する UDP コマンドを示します。

表 10-4　UDP コマンド

コマンド	定義
udp binary send *handle* [*host port*] *message*	ホストおよびポートで指定された宛先へメッセージを含むバイナリ データを送信します。UDP ハンドルが転送終端にすでに接続されている場合は、*host* および *port* 引数は使用されていない可能性があります。UDP ハンドルが接続されていない場合は、これらの任意の引数を使用して、データグラムの宛先を指定してください。
udp bind *handle* **readable** [*script*] **udp bind** *handle* **writable** [*script*]	スクリプトを UDP ハンドルにバインドできます。UDP ハンドルが読み取り可能または書き込み可能になると、**udp bind** コマンドの 3 番めの引数に応じて、スクリプトが評価されます。*script* 引数なしで **udp bind** コマンドを呼び出すことにより、現在 UDP ハンドルにバインドされているスクリプトを読み出すことができます。バインドを削除するには、空のストリングをバインドします。
udp close *handle*	ハンドルに関連付けられた UDP ソケットをクローズします。
udp connect *host port*	UDP のデータグラム ソケットをオープンして、リモート ホスト上のポートに接続します。接続された UDP ソケットは、メッセージを単一の宛先にしか送信できません。このため、通常は接続された UDP ソケットでは各 **udp send** コマンドの宛先アドレスを指定する必要がないので、コードを短縮することができます。コマンドは、UDP ハンドルを返します。
udp info [*handle*]	このコマンドは、*handle* 引数なしですべての既存の UDP ハンドルのリストを返します。有効な UDP ハンドルを指定することにより、UDP ハンドルのステート情報を取得できます。その結果、送信元 IP アドレス、送信元ポート、宛先 IP アドレス、および宛先ポートを含むリストを取得します。
udp open [*port*]	UDP のデータグラム ソケットをオープンして、UDP ハンドルを返します。ソケットは、指定されたポート番号またはポート名にバインドされます。*port* 引数を指定しなかった場合は、未使用のポート番号が使用されます。
udp receive *handle*	ハンドルに関連付けられた UDP ソケットからデータグラムを受信します。このコマンドは、データグラムを受信する準備ができるまでブロックされています。
udp send *handle* [*host port*] *message*	ホストおよびポートで指定された宛先へメッセージを含む ASCII データを送信します。UDP ハンドルが転送終端にすでに接続されている場合は、*host* および *port* 引数は使用されていない可能性があります。UDP ハンドルが接続されていない場合は、これらの任意の引数を使用して、データグラムの宛先を指定してください。

プローブ スクリプト

多様なアプリケーション セットおよびヘルス プローブを使用してネットワークを管理する必要がある場合、CSM では HTTP ヘルス プローブ、TCP ヘルス プローブ、および ICMP ヘルス プローブなどいくつかの特定タイプのヘルス プローブをサポートします。現在の CSM ソフトウェア リリースでサポートされる基本的なヘルス プローブ タイプは、ネットワークの求める特定のプローブ動作をサポートしないことがあります。より柔軟なヘルス プローブ機能をサポートするために、CSM は現在、TCL スクリプトをアップロードし、CSM 上で実行できるようになっています。

プローブに関連付けられたサーバ ファーム内の個々の実サーバに対して、CSM が定期的に実行するスクリプト プローブを作成できます。スクリプトの終了コードに基づいて、実サーバはヘルシー、サスペクト、失敗と判断されます。プローブ スクリプトは、サーバへのネットワーク接続を確立し、サーバにデータを送信し、応答を確認することによって、実サーバの状態をテストします。この TCL スクリプト環境は柔軟性が高いので、プローブ機能が使用できるようになります。

各タイム インターバルを設定すると、CSM の内部スケジューラがヘルス スクリプトをスケジューリングします。単一のプローブ実行が目的となるように、スクリプトを作成してください。**exit** コマンドを使用して、プローブの結果を宣言する必要があります。

一般に、ヘルス スクリプトが実行する動作は次のとおりです。

- IP アドレスに対してソケットをオープンします。
- 1 つまたは複数の要求を送信します。
- 応答を読み取ります。
- 応答を分析します。
- ソケットをクローズします。
- exit 5000（成功）または exit 5001（失敗）を使用することによって、スクリプトを終了します。

新しい **probe** *probe-name* **script** コマンドを使用すると、Cisco IOS ソフトウェアでスクリプト プローブを作成できます。このコマンドによって、従来の CSM ヘルス プローブ サブモード（HTTP、TCP、DNS、SMTP など）と同様のプローブ サブモードが開始されます。プローブ スクリプト サブモードには、従来のプローブ サブモード コマンド（**failed**、**interval**、**open**、**receive**、および **retries**）が含まれます。

さらに、新しい **script** *script-name* コマンドがプローブ スクリプト サブモードに追加されました。このコマンドでは 5 つの引数を使用できます。各引数は、実行時にヘルス プローブ機能の一部としてスクリプトに渡されます。

プローブ スクリプトの記述例

次に、ヘルス スクリプトを使用して HTTP サーバを調べる記述例を示します。

```
Router(config)# !name = HTTP_TEST

# get the IP address of the real server from a predefined global array csm_env
set ip $csm_env(realIP)
set port 80
set url "GET /index.html HTTP/1.0\n\n"

# Open a socket to the server. This creates a TCP connection to the real server
set sock [socket $ip $port]
fconfigure $sock -buffering none -eofchar {}

# Send the get request as defined
puts -nonewline $sock $url;

# Wait for the response from the server and read that in variable line
set line [ read $sock ]

# Parse the response
if { [ regexp "HTTP/1.. (\[0-9\]+) " $line match status  ] } {
  puts "real $ip server response : $status"
}

# Close the socket. Application must close the socket once the
# is over. This allows other applications and tcl scripts to make
# a good use of socket resource. Health monitoring is allowed to open
# only 200 sockets simultaneously.
close $sock

# decide the exit code to return to control module.
# If the status code is OK then script MUST do exit 5000
# to signal successful completion of a script probe.
# In this example any other status code means failure.
# User must do exit 5001 when a probe has failed.
if { $status == 200 } {
    exit 5000
} else {
    exit 5001
}
```

環境変数

ヘルス プローブ スクリプトは、前もって定義された TCL 配列によって、さまざまな設定項目にアクセスできます。この配列の最も一般的な用途は、スクリプトの起動中に疑わしいとされたもの（サスペクト）の、現在の実サーバ IP アドレスを調べることです。

CSM 上でスクリプト プローブが実行されるたびに、csm_env という特殊な配列がスクリプトに渡されます。この配列には、スクリプトが使用する重要なパラメータが格納されます。

（注） ここで使用する環境変数の情報は、プローブ スクリプトのみに適用されます。スタンドアロン スクリプトには適用できません。

表 10-5 に、csm_env 配列のメンバーを示します。

表 10-5　csm_env 配列のメンバー

メンバー名	内容
realIP	サスペクトの IP アドレス
realPort	サスペクトの IP ポート
intervalTimeout	設定されているプローブ インターバル（秒単位）
openTimeout	このプローブに設定されているソケット オープン タイムアウト
recvTimeout	このプローブに設定されているソケット受信タイムアウト
failedTimeout	設定されている障害タイムアウト
retries	設定されている再試行回数
healthStatus	現在のサスペクトのヘルス ステータス

終了コード

プローブ スクリプトは終了コードを使用して、各種の内部状態を示します。終了コード情報は、スクリプトが正常に動作しなかった場合のトラブルシューティングに役立ちます。使用できる終了コードは、**exit 5000** および **exit 5001** だけです。プローブ スクリプトは、スクリプトの終了コードを使用して、実サーバの相対的な正常さとアベイラビリティを示します。スクリプトは exit（5000）を呼び出すことによって、サーバがプローブに正常に応答したことを示します。exit（5001）を呼び出した場合、サーバがヘルス プローブに正しく応答しなかったことを示します。

プローブ スクリプトが失敗して 5001 で終了した場合、対応するサーバは PROBE_FAILED とマーキングされ、一時的にサーバ ファームからディセーブルにされます。CSM はサーバの調査を継続します。プローブが正常に再接続されて 5000 で終了した場合、CSM はサーバのステータスに OPERATIONAL とマーキングして、サーバ ファームからそのサーバを再度イネーブルにします。

exit 5001 のスクリプトに加え、次の状態はスクリプトを失敗させて、PROBE_FAILED（サスペクト）とマーキングされる可能性があります。

- TCL エラー ― スクリプト内に、TCL インタープリタで検出されたエラーが含まれる場合に発生します（構文エラーなど）。構文エラー メッセージは特殊な変数 **erroInfo** に保存されるため、**show mod csm X tech script** コマンドで表示できます。
- スクリプトの停止 ― 無限ループまたはスクリプトが無効な IP アドレスに接続しようとしたことが原因です。各スクリプトは設定された時間内にタスクを完了しなければなりません。スクリプトがタスクを完了できなかった場合、スクリプト コントローラがスクリプトを中止します。サスペクトは暗黙の失敗とみなされます。
- エラー条件 ― 接続のタイムアウトまたはピアによる接続拒否も、暗黙の失敗として扱われる場合に発生します。

表 10-6 に、CSM で使用される終了コードをすべて示します。

表 10-6　CSM の終了コード

終了コード	サスペクトの意味および動作上の影響
5000	サスペクトは正常。ユーザが制御。
5001	サスペクトは失敗。ユーザが制御。
4000	スクリプト打ち切り。ステートの変化はそのときのシステム ステータスによって決まります。システム用に予約。
4001	スクリプト終了。サスペクトは失敗。システム用に予約。
4002	スクリプトはパニック。サスペクトは失敗。システム用に予約。

表 10-6　CSM の終了コード（続き）

終了コード	サスペクトの意味および動作上の影響
4003	スクリプトは内部操作またはシステムコールに失敗。サスペクトは失敗。システム用に予約。
不明	変化なし。

EXIT_MSG 変数

デバッグが目的であれば、特殊な変数 EXIT_MSG にスクリプト デバッグ情報を設定すると効果的です。EXIT_MSG 変数を使用して特定の Cisco IOS **show** コマンドを入力することで、スクリプトの実行ポイントを追跡できます。

次に、EXIT_MSG 変数を使用して、スクリプトの終了点を追跡し、スクリプトが機能しなかった理由を調べる方法を示します。

```
set EXIT_MSG "opening socket"
set s [socket 10.2.0.12 80]
set EXIT_MSG "writing to socket"
puts -nonewline $sock $url
```

EXIT_MSG 変数を調べるには、**show module csm** *slot* **tech script** コマンドを使用します。

次に、終了前に実行した最後のスクリプトが EXIT_MSG であるために、EXIT_MSG が「opening socket」に設定された例を示します。

```
router1# show module csm 4 tech script
SCRIPT CONTROLLER STATS
: ================================================================================
SCRIPT(0xcbcfb50) stat blk(0xcbcfbb0): TCL_test.tclcbcfb50
CMDLINE ARGUMENT:
 curr_id 1 argc 0 flag 0x0::
 type = PROBE
 task_id = 0x0: run_id = 512 ref count = 2
 task_status = TASK_DONE run status = OK
 start time = THU JAN 01 00:15:47 1970
 end time = THU JAN 01 00:17:02 1970
 runs = 1   +0
 resets = 1   +0
 notrel = 0   +0
 buf read err = 0   +0
 killed = 0   +0
 panicd = 0   +0
last exit status= 4000 last Bad status = 4000
Exit status history:
 Status (SCRIPT_ABORT) occured #(1) last@ THU JAN 01 00:17:02 1970
**TCL Controller:
------------------------
tcl cntrl flag = 0x7fffffff
#select(0) close_n_exit(0) num_sock(1)
 MEM TRACK last alloc(0) last size(0) alloc(0) size(0)
 hm_ver (1) flag(0x0) script buf(0xcbf8c00) new script buf(0x0)  lock owner(0x0) sig
taskdel:0 del:0 syscall:0 syslock:0 sig_select  script ptr (0xcbf88f0) id(0)
Config(0xcbcdd78) probe -> 10.1.0.105:80
 tclGlob(0xcbad050) script resource(0xcbcfa28)
#Selects(0) Close_n_exit(0) #Socket(1)
OPEN SOCKETS:
Last erroInfo = couldn't open socket: host is unreachable
    while executing
"socket 10.99.99.99 80 "
    (file "test.tcl" line 2)
Last errorCode = 65
Last panicInfo =
EXIT_MSG = opening socket
```

プローブ スクリプトの実行

プローブ スクリプトを実行するには、スクリプト プローブ タイプを設定し、スクリプト名をプローブ オブジェクトに関連付けます（『*Catalyst 6500 Series Content Switching Module Command Reference*』を参照）。

サーバ ファームおよび仮想サーバにスクリプトをロード、生成、適用してプローブ スクリプトを実行し、結果を表示する手順は、次のとおりです。

ステップ 1　スクリプトをロードします。

```
router1# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
router1(config)# module csm 6

router1(config-module-csm)# script file tftp://192.168.10.102/csmTcl.tcl
Loading csmTcl.tcl from 192.168.10.102 (via Vlan100): !
[OK - 1933 bytes]
```

ステップ 2　スクリプト プローブを作成します。

```
router1(config-module-csm)# probe test1 script
rout(config-slb-probe-script)# script CSMTCL
rout(config-slb-probe-script)# interval 10
rout(config-slb-probe-script)# exit
```

ステップ 3　サーバ ファームおよび仮想サーバにプローブを適用します。

```
router1(config-module-csm)# serverfarm test
router1(config-slb-sfarm)# real 10.1.0.105
router1(config-slb-real)# ins
router1(config-slb-real)# probe test1
router1(config-slb-sfarm)# exit
```

ステップ 4　サーバ ファームを仮想サーバに適用します。

```
router1(config-module-csm)# vserver test
router1(config-slb-vserver)# virtual 10.12.0.80 tcp 80
router1(config-slb-vserver)# serverfarm test
router1(config-slb-vserver)# ins
router1(config-slb-vserver)# exit
```

この時点でスクリプト プローブが設定されます。**show module csm slot tech probe** コマンドを使用して、実行されているスクリプトを確認できます。

ステップ 5　スクリプト プローブを停止します。

```
router1(config-module-csm)# serverfarm test
router1(config-slb-real)# no probe test1
router1(config-slb-sfarm)# exit
```

ここでは、スクリプト コマンドの結果を検証する例を示します。

次に、スクリプト情報を表示する例を示します。

```
router1# show module csm 6 script
CSMTCL, file tftp://192.168.10.102/csmTcl.tcl
  size = 1933, load time = 03:09:03 UTC 01/01/70
```

次に、プローブ スクリプトに関する情報を表示する例を示します。

```
router1# show module csm 6 probe
probe           type    port  interval retries failed  open   receive
------------------------------------------------------------------------
TEST1           script        10       3       300     10     10
router1#
```

次に、特定のプローブ スクリプトに関する詳細情報を表示する例を示します。

```
router1# show module csm 6 probe name TEST1 detail
probe           type    port  interval retries failed  open   receive
------------------------------------------------------------------------
TEST1           script        10       3       300     10     10
 Script: CSMTCL
 real                  vserver         serverfarm      policy          status
 --------------------------------------------------------------------------------
 10.1.0.105:80         TEST1           TEST            (default)       OPERABLE
router1#
```

次に、実サーバのプローブ情報を表示する例を示します。

```
router1# show module csm 6 probe real
  real = 10.1.0.105:80, probe = TEST1, type = script,
    vserver = TEST, sfarm = TEST
    status = FAILED, current = 03:26:04 UTC 01/01/70,
    successes = 1, last success = 03:15:33 UTC 01/01/70,
    failures = 4, last failure = 03:26:04 UTC 01/01/70,
    state = Unrecognized or invalid response
    script CSMTCL
    last exit code = 5001
```

プローブ スクリプトのデバッグ

スクリプト プローブのデバッグ方法は次のとおりです。

- verbose モードで実行しているスクリプトに TCL の **puts** コマンドを使用します。

  verbose モードで **puts** コマンドを使用すると、各プローブ サスペクトは CSM コンソールにストリングを出力します。システムで実行しているサスペクトが多いと、リソースもそれだけ必要になるため、CSM コンソールがハングする可能性があります。この機能は、システムに設定されたサスペクトが 1 つである場合にのみイネーブルにすることを推奨します。

- スクリプトで特殊変数 EXIT_MSG を使用します。

  各プローブ サスペクトに自身の EXIT_MSG 変数が含まれています。この変数によりスクリプトのステータスを追跡して、プローブの状態を確認できます。

  次に、スクリプトの EXIT_MSG 変数の使用例を示します。

  ```
  set EXIT_MSG "before opening socket"
  set s [ socket $ip $port]
  set EXIT_MSG " before receive string"
  gets $s
  set EXIT_MSG "before close socket"
  close $s
  ```

  メッセージ受信時にプローブ サスペクトが失敗した場合、ストリングを受信する前に EXIT_MSG = を確認する必要があります。

- **show module csm** *slot* **probe real** [*ip*] コマンドを使用します。

  このコマンドは、システム内で現在アクティブなプローブ サスペクトを表示します。

```
router1# show module csm 6 probe real
  real = 10.1.0.105:80, probe = TEST1, type = script,
    vserver = TEST, sfarm = TEST
    status = FAILED, current = 04:06:05 UTC 01/01/70,
    successes = 1, last success = 03:15:33 UTC 01/01/70,
    failures = 12, last failure = 04:06:05 UTC 01/01/70,
    state = Unrecognized or invalid response
    script CSMTCL
    last exit code = 5001
```

（注） 最後の終了コードには、表 10-6（p.10-10）に示す終了コードのうち 1 つが表示されます。

- **show module csm** *slot* **tech probe** コマンドを使用します。

  このコマンドは現在のプローブ ステータスを表示します（標準およびスクリプト プローブの両方）。

```
router1# show module csm 6 tech probe

Software version: 3.2(1)
------------------------------------------------------------
------------------ Health Monitor Statistics ------------
------------------------------------------------------------
 Probe templates: 1
Suspects created: 1
    Open Sockets in System : 8 / 240
    Active Suspect(no ICMP): 0 / 200
    Active Script Suspect  : 0 / 50
    Num events  : 1

Script suspects: 1
  Healthy suspects: 0
Failures suspected: 0
Failures confirmed: 1

Probe attempts:         927  +927
Total recoveries:         3  +3
Total failures:           6  +6
Total Pending:            0  +0
```

- **show module csm** *slot* **tech script** コマンドを使用して、最後の終了ステータス、固定変数、errorInfo、および EXIT_MSG の出力を検出します。

```
router1# show module csm 6 tech script
SCRIPT(0xc25f7e0) stat blk(0xc25f848): TCL_csmTcl.tclc25f7e0
CMDLINE ARGUMENT:
 curr_id 1 argc 0 flag 0x0::
 type = PROBE
 task_id = 0x0: run_id = 521 ref count = 2
 task_status = TASK_DONE run status = OK
 start time = THU JAN 01 03:51:04 1970
 end time = THU JAN 01 03:51:04 1970

 runs = 13   +11
 resets = 13   +11
 notrel = 0   +0
 buf read err = 1   +1
 killed = 0   +0
 panicd = 0   +0

last exit status= 5001 last Bad status = 5001

Exit status history:

**TCL Controller:
------------------------
tcl cntrl flag = 0x7fffffff
#select(0) close_n_exit(0) num_sock(2)
 MEM TRACK last alloc(0) last size(0) alloc(0) size(0)
 hm_ver (3) flag(0x0) script buf(0xc25ad80) new script buf(0xc25ad80)
 lock owner(0x0) sig taskdel:0 del:0 syscall:0 syslock:0 sig_select
 script ptr (0xc25f038) id(0)
Config(0xc2583d8) probe -> 10.1.0.105:80
 tclGlob(0xc257010)
SCRIPT RESOURCE(0xc25af70)-------
#Selects(0) Close_n_exit(0) #Socket(2)
OPEN SOCKETS:

Persistent Variables

--------------------
   x = 11

Last erroInfo =

Last errorCode =
Last panicInfo =
EXIT_MSG = ping failed : invalid command name "ping"
```

最後の終了ステータスには、終了コード番号（表 10-6[p.10-10] に示す）が表示されます。

固定変数情報は、**gset varname value** コマンドで設定されます（表 10-3[p.10-5] を参照）。

erroInfo 変数は TCL コンパイラで生成されたエラーのリストです。スクリプトに TCL のランタイム エラーが含まれている場合、TCL インタープリタはスクリプトの実行を停止して、erroInfo 変数のエラー情報を保存します。

EXIT_MSG（「EXIT_MSG 変数」[p.10-11] を参照）は、失敗の恐れがある各プローブの詳細なデバッグ情報を表示します。出力が非常に長くなる可能性があるため、次の例のようにキーワードで最初にフィルタリングすることもできます。

```
router1# show module csm slot tech script inc keyword
```

スタンドアロン スクリプト

スタンドアロン スクリプトは、CSM にロードして実行する一般的な TCL スクリプトです。スタンドアロン スクリプトはプローブ スクリプトのように設定してサーバ ファームに適用しないため、CSM の定期的な実行タスクとしてのスケジューリングは行われません。タスクを実行するには、**script task** コマンドを使用する必要があります。

csm_env 環境変数はスタンドアロン スクリプトには適用されません。ただし、プローブ スクリプトの終了コードにスタンドアロン スクリプトに対する特別な意味が含まれていない場合、**exit** コマンドを使用できます。

スタンドアロン スクリプトの記述例

次に、一般的な TCL スクリプトの記述例を示します。

```
#!name = STD_SCRIPT
set gatewayList "1.1.1.1 2.2.2.2"
foreach gw $gatewayList {
    if { ![ ping $gw ]  } {
    puts "-WARNING : gateway $gw is down!!"
  }
}
```

スタンドアロン スクリプトの実行

スタンドアロン スクリプトはスクリプト プローブと異なり、タスクを 1 度だけ実行する TCL スクリプトです。スクリプトのロードが完了すると、実行が終了します。スタンドアロン スクリプトはタスクとしてスクリプトが設定されていないかぎり、定期的に CSM が実行することはありません。**script file** コマンドをスタートアップ コンフィギュレーションに組み込むと、CSM の起動時にコマンドが実行されます。スクリプトは CSM の稼働中、引き続き実行されます。

スタンドアロン スクリプトを実行するには、次の手順を実行します。

ステップ 1　スクリプトをロードします。

```
router1# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
router1(config)# module csm 6
router1(config-module-csm)# script file tftp://192.168.10.102/stdcsm.tcl
Loading stdcsm.tcl from 192.168.10.102 (via Vlan100): !
[OK - 183 bytes]
```

ステップ 2　スタンドアロン タスクとしてスクリプトを実行します。

```
router1(config-module-csm)# script task 1 STD_SCRIPT
```

ステップ 3　スクリプトを再度実行します。

古いタスクを削除して、次のように再度実行できます。

```
router1(config-module-csm)# no script task 1 STD_SCRIPT
router1(config-module-csm)# script task 1 STD_SCRIPT
```

次のように新しいタスク ID を指定することで、新規タスクを開始することもできます。

```
router1(config-module-csm)# script task 2 STD_SCRIPT
```

ステップ 4　スクリプトを停止します。

```
router1(config-module-csm)# no script task 1 STD_SCRIPT
```

ステップ 5　スクリプトのステータスを表示するには、**show** コマンドを使用します。

```
router1#sh mod csm 6 script
STD_SCRIPT, file tftp://192.168.10.102/stdcsm.tcl
router1#sh mod csm 6 script task

task  script                          runs       exit code  status
------------------------------------------------------------------------------
1     STD_SCRIPT                      1          4000       Not Ready
2     STD_SCRIPT                      1          4000       Not Ready
```

特定の実行スクリプトに関する情報を表示するには、**show module csm** *slot* **script task index** *script-index* **detail** または **show module csm** *slot* **script name** *script-name* **code** コマンドを使用します。

スタンドアロン スクリプトのデバッグ

スタンドアロン スクリプトのデバッグは、プローブ スクリプトのデバッグと類似しています（「プローブ スクリプトのデバッグ」[p.10-13] を参照）。複数のスレッドを実行しても問題は起きないので、スクリプトで **puts** コマンドを使用して、デバッグに役立てることができます。

TCL スクリプトの FAQ

ここでは、CSM の TCL スクリプトに関するよくある質問（FAQ）を取り上げます。

- システム リソースはどのように使用されますか？

  VxWorks サポート アプリケーションには、ファイル ディスクリプタが 255 個あり、それが標準入力、標準出力、あらゆるソケット接続（受信または送信）など、すべてのアプリケーション間で分配されます。スタンドアロン スクリプトを作成する場合は、いつソケットをオープンするかについて、慎重に配慮する必要があります。リソース不足を防ぐために、動作の完了後、ただちにソケットをクローズすることを推奨します。ヘルス モニタリング モジュールは、活発に実行しているスクリプトの数を制御することによって、オープン ソケット数を制御します。スタンドアロン スクリプトには、このような制御がありません。

  メモリは考慮事項ではあるものの、大きな制約をもたらすことはありません。モジュールには通常、十分なメモリが装備されているからです。スクリプトごとに 128 KB のスタックを 1 つずつ使用し、残りのメモリが実行時にスクリプトによって割り当てられます。

  スクリプトの実行中もシステムのリアルタイム特性がある程度一定に保てるように、スクリプト タスクにはシステムで最下位のプライオリティが与えられます。スクリプトのプライオリティが低いということは残念ながら、システムが TCL 以外の動作で忙しい場合に、どの TCL スレッドも完了に時間がかかることを意味します。このような状況では、一部のヘルス スクリプトが打ち切られ、未完了のスレッドにエラーのマークが設定されることがあります。スクリプトがエラーにならないように、すべてのスクリプト プローブで再試行値を 2 以上に設定する必要があります。可能なかぎり、固有の CSM プローブ（HTTP、DNS など）を使用してください。スクリプトのヘルス プローブは、サポート外のアプリケーションをサポートする場合に使用してください。ヘルス プローブの目的は、高速化ではなく、機能を提供することです。

  TCL は同期と非同期の両方のソケット コマンドをサポートします。非同期のソケット コマンドは、実接続を待たずに、ただちに戻ります。非同期バージョンのスクリプトを内部で実装するには、この種のコマンドごとに多数のシステムコールを使用する、非常に複雑なコード パスが必要になります。一般にこのような状況は、他のコマンドによるシステム コールの処理中にいくつかのクリティカル リソースが待機する原因となるので、システム速度を低下させます。どうしても避けられない場合以外、スクリプト プローブに非同期ソケットを使用することは推奨できません。ただし、スタンドアロン システムでこのコマンドを使用することは可能です。

- 設定したプローブが実行されているかどうかを知る方法は？

  ネットワークの実サーバ側で Sniffer を実行できます。また、次の **show** コマンドを使用して、プローブが CSM 上で実行されているかどうかを判断できます。

  — プローブが実行されていれば、この例で示すようにプローブを試行した回数が増加していきます。

```
router1# show module csm 6 tech probe
router1#sh mod csm 6 tech probe
Software version: 3.2(1)
------------------------------------------------------------
------------------ Health Monitor Statistics -----------------
------------------------------------------------------------
 Probe templates: 8
Suspects created: 24
    Open Sockets in System : 10 / 240
    Active Suspect(no ICMP): 2 / 200
    Active Script Suspect  : 2 / 50
    Num events  : 24
Script suspects: 24
  Healthy suspects: 16
Failures suspected: 0
Failures confirmed: 8
Probe attempts:        321  +220
Total recoveries:       16  +0
Total failures:          8  +2
Total Pending:           0  +0
```

— プローブが実行されていれば、この例で示すように成功または失敗のカウントが増加します。

```
router1# show module csm 6 probe real
 real = 10.12.0.108:50113, probe = SCRIPT2_2, type = script,
   vserver = SPB_SCRIPT2, sfarm = SCRIPT2_GOOD, policy = SCRIPT2_GOOD,
   status = OPERABLE, current = 22:52:24 UTC 01/04/70,
   successes = 18, last success = 22:52:24 UTC 01/04/70,
   failures = 0, last failure = 00:00:00 UTC 01/01/70,
   state = Server is healthy.
   script httpProbe2.tcl GET /yahoo.html html 1.0 0
   last exit code = 5000
 real = 10.12.0.107:50113, probe = SCRIPT2_2, type = script,
   vserver = SPB_SCRIPT2, sfarm = SCRIPT2_GOOD, policy = SCRIPT2_GOOD,
   status = OPERABLE, current = 22:52:42 UTC 01/04/70,
   successes = 19, last success = 22:52:42 UTC 01/04/70,
   failures = 0, last failure = 00:00:00 UTC 01/01/70,
   state = Server is healthy.
   script httpProbe2.tcl GET /yahoo.html html 1.0 0
   last exit code = 5000
```

また、リセット（RST）の代わりに FIN を使用してソケットをクローズすることもできます。

- リモート ホストが到達不能の場合、UDP プローブが実サーバを PROBE_FAIL ステートにできない理由は？

  UDP プローブは「icmp port unreachable」メッセージを受信して、サーバに PROBE_FAIL をマーキングする必要があります。リモート ホストがダウンした場合、または応答しない場合、UDP プローブは ICMP メッセージを受信しないために、そのプローブのパケットは紛失したとみなされ、サーバは正常の状態にあると判断されます。

  UDP プローブは Raw UDP プローブなので、CSM はプローブの応答ペイロードに単一のバイトを使用します。CSM は UDP アプリケーションから意味のある応答がくることを想定していません。CSM は ICMP 到達不能メッセージを使用して、UDP アプリケーションが到達可能かどうかを判断します。

  受信タイムアウトで ICMP 到達不能の応答がない場合、CSM はプローブが正常に実行されていると判断します。実サーバの IP インターフェイスがダウンまたは切断された場合、UDP プローブは自身で UDP アプリケーションが到達不能にあることを判断できません。指定のサーバの UDP プローブのほかに ICMP プローブを設定する必要があります。

  回避策：常に ICMP を UDP のプローブ タイプで設定します。

- ダウンロード可能なサンプル スクリプトはどこにありますか？

  サンプル スクリプトを使用して TCL 機能をサポートできます。その他のカスタム スクリプトも使用できますが、これらのサンプル スクリプトはシスコシステムズの TAC がサポートしています。サンプル スクリプトのファイルには、次の URL からアクセスしてください。

  http://www.cisco.com/cgi-bin/tablebuild.pl/cat6000-intellother

  スクリプト ファイルの名前は、c6slb-script.3-3-1.tcl です。

- TCL スクリプト情報はどこにありますか？

  TCL 8.0 のコマンド リファレンスには、次の URL からアクセスできます。

  http://www.tcl.tk/man/tcl8.0/TclCmd/contents.html

  TCL UDP コマンド リファレンスには、次の URL からアクセスできます。

  http://wwwhome.cs.utwente.nl/~schoenw/scotty/

CHAPTER 11

ファイアウォール ロードバランシングの設定

この章では、ファイアウォール ロードバランシングの設定方法について説明します。

ファイアウォール ロードバランシングを使用すると、接続単位で複数のファイアウォールにトラフィックを分散させることによって、ファイアウォールの保護を拡張することができます。特定の接続に属すパケットはすべて、同じファイアウォールに送られなければなりません。ファイアウォールが個々のパケットについて、ファイアウォールのインターフェイスを通過することを許可または拒否します。

ファイアウォールの機能

ファイアウォールは、ネットワークの 2 つの部分（たとえば、インターネットとイントラネットなど）の間に物理的な境界を形成します。ファイアウォールは一方（インターネット）からパケットを受け付け、そのパケットを他方（イントラネット）に送り出します。ファイアウォールはパケットを変更してから渡すことも、そのまま送り出すこともできます。ファイアウォールがパケットを拒否する場合、通常はパケットを廃棄し、パケット廃棄をイベントとして記録します。

セッションが確立され、パケット フローが開始されると、ファイアウォールはそのファイアウォールに設定されているポリシーに従って、フロー内の各パケットをモニタするか、またはモニタしないでフローを流し続けます。

この章の内容は、次のとおりです。

ファイアウォールのタイプ

ファイアウォールの基本的なタイプは、次のとおりです。

- 標準ファイアウォール
- ステルス ファイアウォール

標準ファイアウォールは、ネットワーク上でその存在が認識されます。装置として宛先になれるように、また、ネットワーク上の他の装置によって認識されるように、IP アドレスが割り当てられます。

ステルス ファイアウォールは、ネットワーク上でその存在が認識されません。したがって、IP アドレスは割り当てられず、宛先になることも、ネットワーク上の他の装置に認識されることもありません。ネットワークにとって、ステルス ファイアウォールは配線の一部です。

どちらのファイアウォール タイプも、（ネットワークの保護された側と保護されていない側の間で）双方向に流れるトラフィックを検証し、ユーザが定義したポリシー セットに基づいて、パケットを受け付けるか、または拒否します。

CSM によるファイアウォールへのトラフィック分散

Content Switching Module（CSM; コンテント スイッチング モジュール）は、サーバ ファーム内に設定されている装置にトラフィックの負荷を分散させます。対象となる装置はサーバ、ファイアウォール、またはエイリアス IP アドレスを含め、IP アドレス指定が可能なあらゆるオブジェクトです。CSM は装置タイプに関係なく、ロードバランス アルゴリズムを使用して、サーバ ファーム内で設定されている装置間でトラフィックをどのように分散させるかを決定します。

（注） 上位レイヤのロードバランス アルゴリズムとサーバ アプリケーション間の相互作用を考えると、ファイアウォールが含まれるサーバ ファームにレイヤ 3 ロードバランシングを設定することを推奨します。

サポート対象のファイアウォール

CSM は、標準ファイアウォールまたはステルス ファイアウォールにトラフィックの負荷を分散させることができます。

標準ファイアウォールでは、CSM がサーバにトラフィックを分散させる場合と同様、単一またはペアの CSM が固有の IP アドレスを持つファイアウォール間でトラフィックを分散させます。

ステルス ファイアウォールの場合、CSM はステルス ファイアウォール経由のパスを提供する別の CSM 上の、固有の VLAN（仮想 LAN）エイリアス IP アドレスを持つインターフェイス間でトラフィックを分散させます。ステルス ファイアウォールは、その VLAN 上を双方向に流れるあらゆるトラフィックがファイアウォールを通過するように設定します。

ファイアウォールに対するレイヤ 3 ロードバランシング

CSM がトラフィックの負荷をファイアウォールに分散させる場合、CSM はサーバにトラフィックの負荷を分散させる場合と同じ機能を実行します。ファイアウォールに対するレイヤ 3 ロードバランシングを設定する手順は、次のとおりです。

ステップ 1　ファイアウォールの両側にサーバ ファームを作成します。

ステップ 2　サーバファーム サブモードで、プレディクタの **hash address** コマンドを入力します。

ステップ 3　ファイアウォール宛のトラフィックを受け付ける仮想サーバに、サーバ ファームを割り当てます。

（注） ファイアウォールに対するレイヤ 3 ロードバランシングを設定するときには、正方向で送信元 Network Address Translation（NAT; ネットワーク アドレス変換）を、逆方向で宛先 NAT を使用します。

ファイアウォール構成タイプ

CSM は、2 種類のファイアウォール構成をサポートします。

- デュアル CSM 構成 — 2 つの CSM の間にファイアウォールを配置します。ファイアウォールは一方の CSM からトラフィックを受け付け、他方の CSM に送ってサーバへの負荷分散を図るか、または要求側装置に戻します。
- シングル CSM 構成 — ファイアウォールは CSM からトラフィックを受け付け、同じ CSM に送り返してサーバへの負荷分散を図るか、または要求側装置にトラフィックを戻します。

ファイアウォール用 IP リバーススティッキ

CSM は現在、固定（sticky）接続をサポートしています。固定接続によって、同じクライアントから発信された異なる 2 つのデータ フローが、同じ宛先にロードバランスされます。

ロードバランスを図る宛先は、実サーバになることがよくあります。ファイアウォール、キャッシュ、またはその他のネットワーキング装置になることもあります。固定接続は、ロードバランス対象のアプリケーションを正しく動作させるために必要です。これらのアプリケーションは、同一クライアントから特定のサーバへの複数の接続を利用します。ある接続で転送された情報が、別の接続で転送された情報の処理を左右する場合があります。

IP スティッキ インサート（sticky intert）機能は、同一クライアントから同一サーバへの新しい接続のバランスを図るために設定します。「ファイアウォール用リバーススティッキの設定」（p.11-27）を参照してください。この機能は、FTP データ チャネル、ストリーミング UDP データ チャネルなど、バディ（buddy）接続の場合に特に重要です。

CSM のファイアウォール設定

CSM がサポートできるファイアウォール設定は、次のとおりです。

- デュアル CSM 構成のステルス ファイアウォール（図 11-1）
- デュアル CSM 構成の標準ファイアウォール（図 11-2）
- シングル CSM 構成の標準ファイアウォール（図 11-3）
- デュアル CSM 構成の混在型（ステルスおよび標準）ファイアウォール（図 11-4）

図 11-1 では、トラフィックはファイアウォールを通過し、双方向でフィルタリングされます。図は、インターネットからイントラネットへの流れを示しています。イントラネットへの経路では、CSM A が VLAN5、6、および 7 にトラフィックを分散させ、ファイアウォール経由で CSM B に送ります。インターネットへの経路では、CSM B が VLAN 15、16、および 17 にトラフィックを分散させ、ファイアウォール経由で CSM A に送ります。CSM A はサーバ ファームで CSM B の VLAN エイリアスを使用し、CSM B はサーバ ファームで CSM A の VLAN エイリアスを使用します。

図 11-1　ステルス ファイアウォールの設定（デュアル CSM 専用）

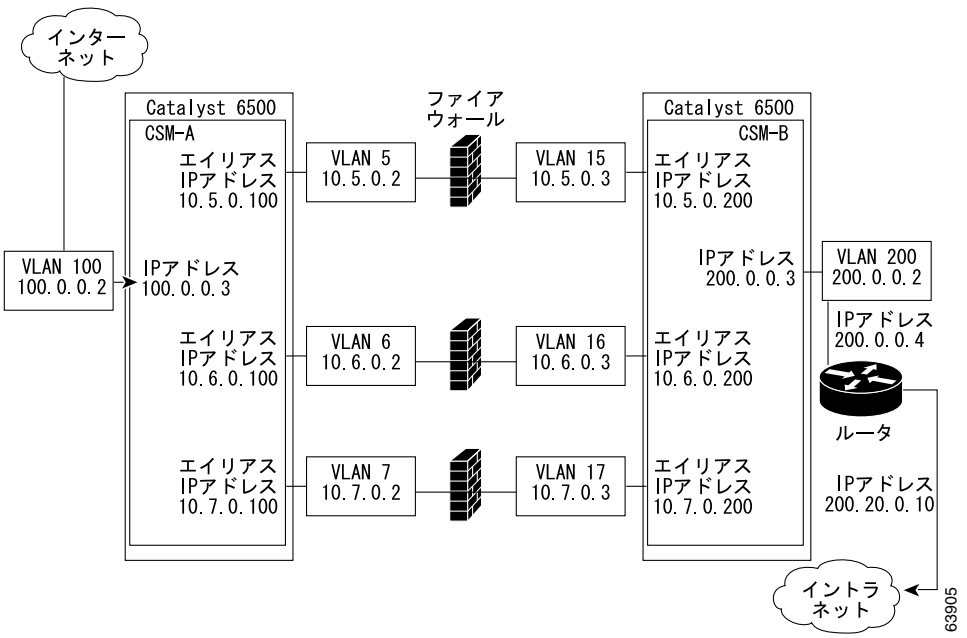


図 11-2 では、トラフィックはファイアウォールを通過し、双方向でフィルタリングされます。図は、インターネットからイントラネットへの流れを示しています。VLAN 11 および 111 が同じサブネットにあり、VLAN 12 および 112 が同じサブネットにあります。

図 11-2　標準ファイアウォールの設定（デュアル CSM）

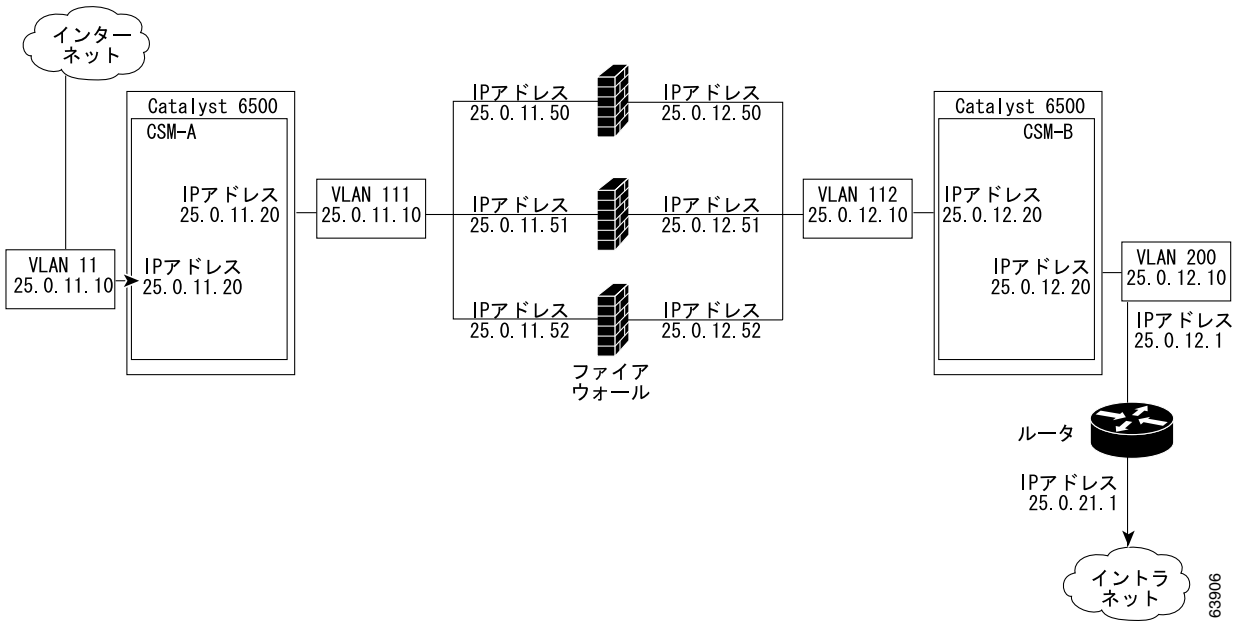


図 11-3 では、トラフィックはファイアウォールを通過し、双方向でフィルタリングされます。図に示されているのは、インターネットからイントラネットへの流れだけです。VLAN 11 および 111 は同じサブネットにあります。VLAN 12 および 112 は同じサブネットにあります。

図 11-3　標準ファイアウォールの設定（シングル CSM）

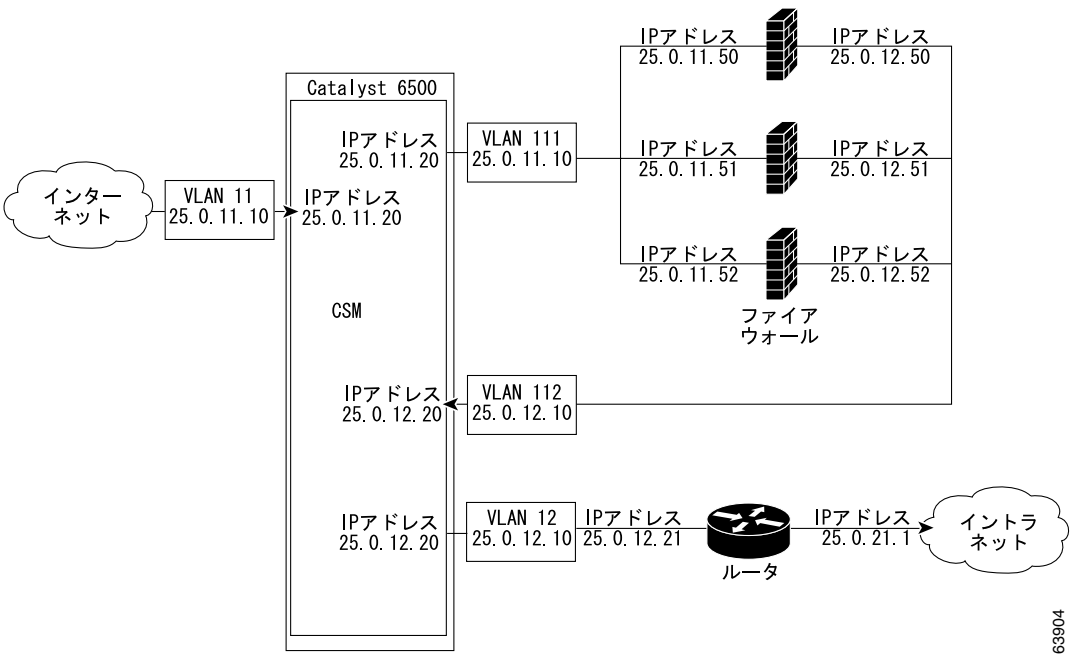
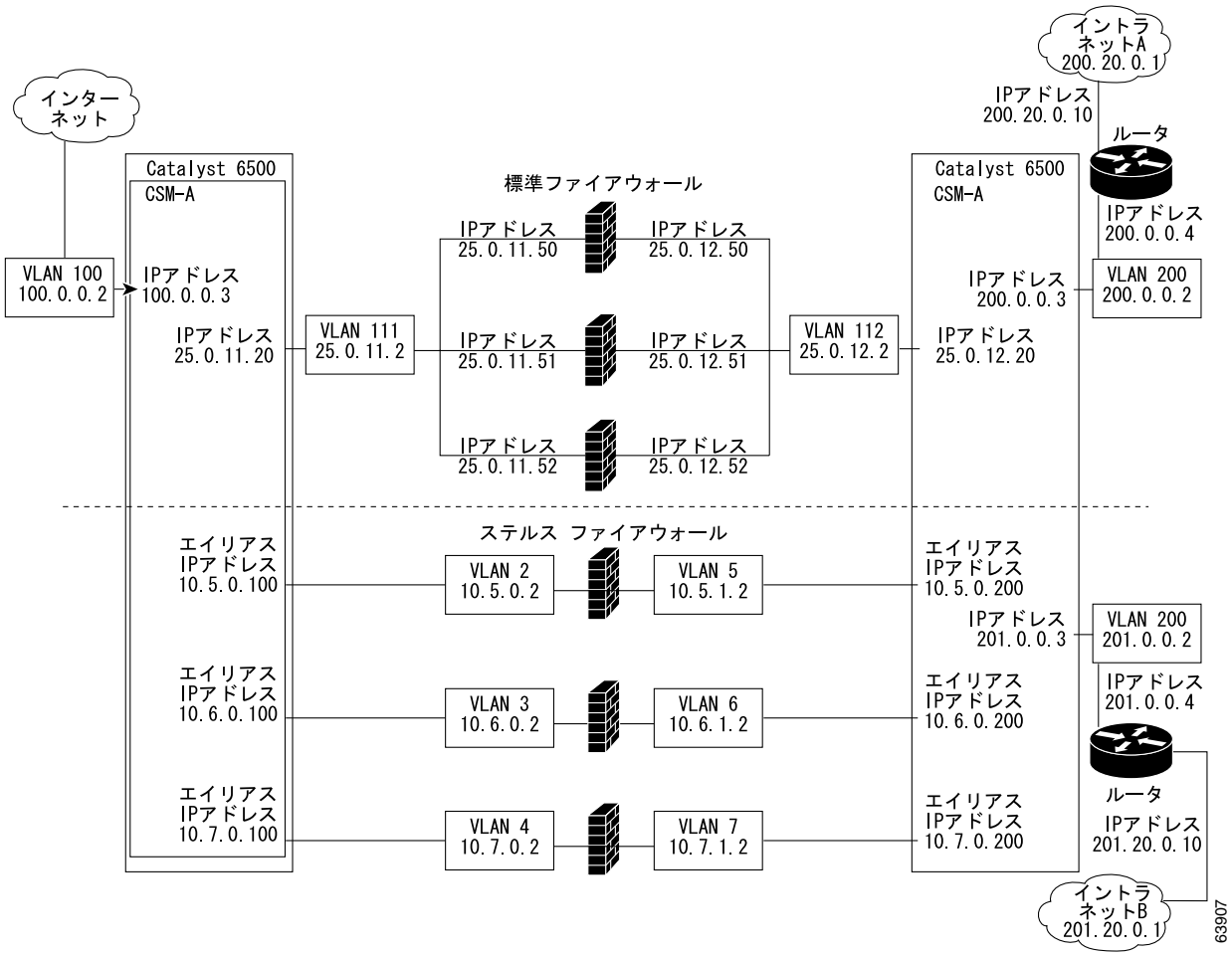


図 11-4 では、トラフィックは標準ファイアウォールとステルス ファイアウォールの両方を通過し、双方向でフィルタリングされます。図は、インターネットからイントラネットへの流れを示しています。VLAN 5、6、および 7 は CSM A および CSM B 間で共有されます。イントラネットへの経路上で、CSM A は VLAN 5、6、および 7 間でトラフィックを分散させ、ファイアウォール経由で CSM B に送ります。イントラネットへの経路上で、CSM B は VLAN 5、6、および 7 間でトラフィックを分散させ、ファイアウォール経由で CSM A に送ります。

図 11-4　ステルスおよび標準ファイアウォールの混在型ファイアウォール設定（デュアル CSM 専用）



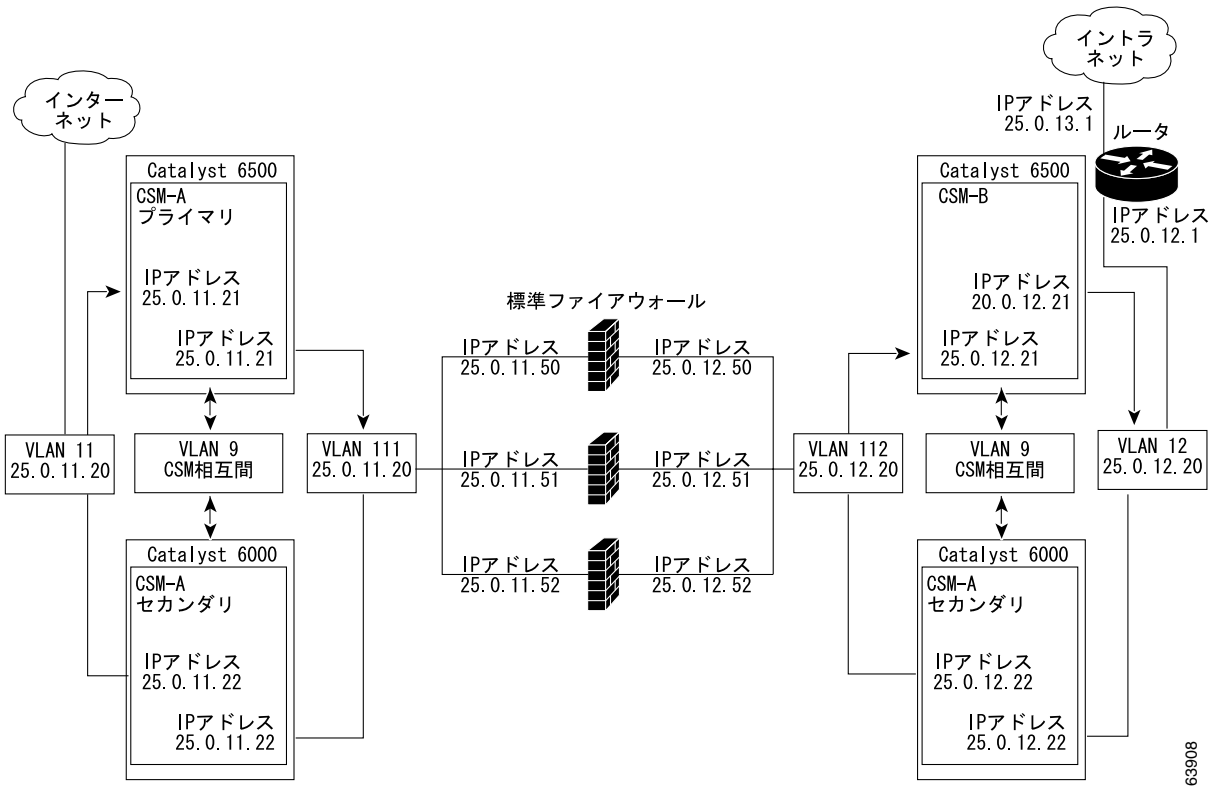
フォールトトレラントな CSM ファイアウォール設定

CSM は、次の構成でフォールトトレランスをサポートします。

- フォールトトレラント デュアル CSM 構成のステルス ファイアウォール
- フォールトトレラント デュアル CSM 構成の標準ファイアウォール
- フォールトトレラント シングル CSM 構成の標準ファイアウォール
- フォールトトレラント デュアル CSM 構成の混在型ファイアウォール（ステルスおよび標準）

図 11-5 では、トラフィックはファイアウォールを通過し、双方向でフィルタリングされます。図に示されているのは、プライマリ CSM を通過する、インターネットからイントラネットへの流れだけです。VLAN 11 および 111 は同じサブネットにあります。VLAN 12 および 112 は同じサブネットにあります。

図 11-5　フォールトトレラントな標準ファイアウォールの設定（デュアル CSM）



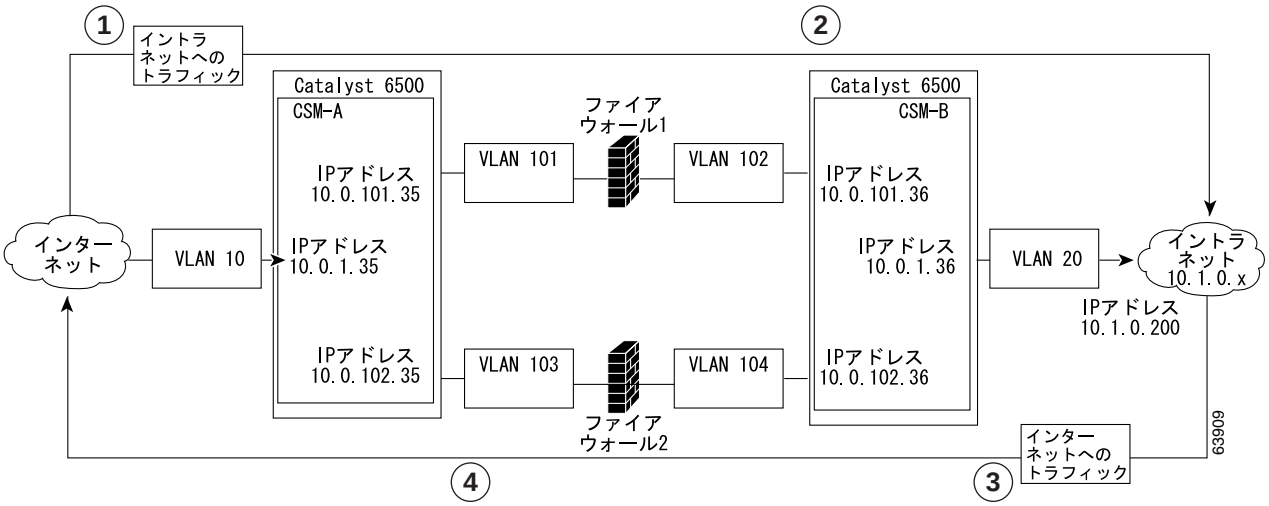
ステルス ファイアウォール ロードバランシングの設定

ここでは、ステルス ファイアウォール用にファイアウォール ロードバランシングを設定する方法について説明します。

ステルス ファイアウォールの設定

ステルス ファイアウォール設定では、ファイアウォールは 2 つの異なる VLAN に接続し、接続先 VLAN の IP アドレスを指定して設定します（図 11-6 を参照）。

図 11-6　ステルス ファイアウォールの設定例



位置	トラフィックの方向	入口	出口
1	イントラネットへ	VLAN 10	VLAN 101 および 103
2	イントラネットへ	VLAN 101 および 103	VLAN 20
3	インターネットへ	VLAN 20	VLAN 102 および 104
4	インターネットへ	VLAN 101 および 103	VLAN 10

図 11-6 では、2 つの標準ファイアウォール（ファイアウォール 1 およびファイアウォール 2）が 2 つの CSM（CSM A および CSM B）の間にあります。

（注）　ステルス ファイアウォールは VLAN 上にアドレスがありません。

インターネットからイントラネットへの経路上で、トラフィックはファイアウォールの保護されていない側から入り、別個の VLAN（VLAN 101 および VLAN 103）を通過し、ファイアウォールの保護された側から出て別個の VLAN（VLAN 102 および VLAN 104）を通過します。イントラネットからインターネットへの経路では、この流れが逆になります。VLAN はインターネット（VLAN 10）およびイントラネット（VLAN 20）への接続も可能にします。

ステルスの設定では、CSM A および CSM B がトラフィックの負荷を分散させてファイアウォールに通します。

ステルス ファイアウォールの設定例

ステルス ファイアウォールの設定例では、2 つの CSM（CSM A および CSM B）をそれぞれ別個の Catalyst 6500 シリーズ スイッチに搭載しています。

（注）　ステルス ファイアウォールの設定では、各 CSM をそれぞれ別個の Catalyst 6500 シリーズ スイッチに搭載する必要があります。

ここでは、CSM A および CSM B 用に、ステルス ファイアウォール コンフィギュレーションを作成する手順について説明します。

CSM A の設定（ステルス ファイアウォールの例）

標準の設定例を作成するには、CSM A に対して次の作業が必要です。

- スイッチ A 上での VLAN の作成（p.11-9）
- CSM A 上での VLAN の設定（p.11-9）
- CSM A 上でのサーバ ファームの設定（p.11-10）
- CSM A 上での仮想サーバの設定（p.11-11）

（注）　設定作業は CSM A でも CSM B でも同じですが、手順、入力するコマンド、およびパラメータが異なります。

スイッチ A 上での VLAN の作成

スイッチ A 上で 2 つの VLAN を作成する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Switch-A(config)# vlan`	VLAN モードを開始します[1]。
ステップ 2	`Switch-A(vlan)# vlan 10`	VLAN 10 を作成します[2]。
ステップ 3	`Switch-A(vlan)# vlan 101`	VLAN 101 を作成します[3]。
ステップ 4	`Switch-A(vlan)# vlan 103`	VLAN 103 を作成します[4]。

1. この作業は、CSM A が搭載されたスイッチのコンソールで行います。
2. VLAN 10 は CSM A をインターネットに接続します。
3. VLAN 101 は、ファイアウォール 1 経由で CSM B に接続します。
4. VLAN 103 は、ファイアウォール 2 経由で CSM B に接続します。

CSM A 上での VLAN の設定

3 つの VLAN を設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Switch-A(config)# module csm 5`	マルチモジュール コンフィギュレーション モードを開始し、CSM A がスロット 5 に搭載されていることを指定します。

	コマンド	目的
ステップ 2	`Switch-A(config-module-csm)# vlan 10 client`	設定対象の VLAN として VLAN 10 を指定し、クライアント VLAN であることを指定し、VLAN コンフィギュレーション モードを開始します。
ステップ 3	`Switch-A(config-slb-vlan-client)# ip address 10.0.1.35 255.255.255.0`	VLAN 10 の IP アドレスおよびネットマスクを指定します。
ステップ 4	`Switch-A(config-slb-vlan-client)# alias 10.0.1.30 255.255.255.0`	VLAN 10 用のエイリアス IP アドレスおよびネットマスクを指定します[1]。
ステップ 5	`Switch-A(config-slb-vlan-client)# exit`	VLAN コンフィギュレーション モードに戻ります。
ステップ 6	`Switch-A(config-module-csm)# vlan 101 server`	設定対象の VLAN として VLAN 101 を指定し、サーバ VLAN であることを指定し、VLAN コンフィギュレーション モードを開始します。
ステップ 7	`Switch-A(config-slb-vlan-server)# ip address 10.0.101.35 255.255.255.0`	VLAN 101 の IP アドレスおよびネットマスクを指定します。
ステップ 8	`Switch-A(config-slb-vlan-server)# alias 10.0.101.100 255.255.255.0`	VLAN 101 用のエイリアス IP アドレスおよびネットマスクを指定します[1]。
ステップ 9	`Switch-A(config-slb-vlan-server)# exit`	VLAN コンフィギュレーション モードに戻ります。
ステップ 10	`Switch-A(config-module-csm)# vlan 103 server`	設定対象の VLAN として VLAN 103 を指定し、サーバ VLAN であることを指定し、VLAN コンフィギュレーション モードを開始します。
ステップ 11	`Switch-A(config-slb-vlan)# ip address 10.0.102.35 255.255.255.0`	VLAN 103 の IP アドレスおよびネットマスクを指定します。
ステップ 12	`Switch-A(config-slb-vlan)# alias 10.0.102.100 255.255.255.0`	VLAN 103 用のエイリアス IP アドレスおよびネットマスクを指定します[1]。

1. このステップで、ロードバランシングの決定に使用する、CSM B のターゲットを特定します。

CSM A 上でのサーバ ファームの設定

（注） CSM B の IP アドレスを INSIDE-SF サーバ ファームで実サーバとして指定するので、CSM A は CSM B への経路上にある 2 つのファイアウォール間で負荷を分散させます。

CSM A 上で 2 つのサーバ ファームを設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Switch-A(config)# module csm 5`	マルチモジュール コンフィギュレーション モードを開始し、CSM A がスロット 5 に搭載されていることを指定します。
ステップ 2	`Switch-A(config-module-csm)# serverfarm FORWARD-SF`	FORWARD-SF[1] サーバ ファーム（実際にはフォワーディング ポリシー）を作成して名前を指定し、サーバ ファーム コンフィギュレーション モードを開始します。
ステップ 3	`Switch-A(config-slb-sfarm)# no nat server`	サーバの IP アドレスおよびポート番号の NAT をディセーブルにします[2]。

	コマンド	目的
ステップ 4	`Switch-A(config-slb-sfarm)# predictor forward`	ロードバランス アルゴリズムではなく、内部ルーティング テーブルに従って、トラフィックを転送します。
ステップ 5	`Switch-A(config-slb-sfarm)# exit`	マルチモジュール コンフィギュレーション モードに戻ります。
ステップ 6	`Switch-A(config-module-csm)# serverfarm TO-INSIDE-SF`	（実サーバではなくエイリアス IP アドレスを指定する）INSIDE-SF[3] サーバ ファームを作成して名前を指定し、サーバ ファーム コンフィギュレーション モードを開始します。
ステップ 7	`Switch-A(config-slb-sfarm)# no nat server`	サーバの IP アドレスおよびポート番号の NAT をディセーブルにします[4]。
ステップ 8	`Switch-A(config-slb-sfarm)# predictor hash address source 255.255.255.255`	送信元 IP アドレスに基づくハッシュ値を使用して、サーバを選択します[5]。
ステップ 9	`Switch-A(config-slb-sfarm)# real 10.0.101.200`	ファイアウォール 1 への経路上にある、CSM B のエイリアス IP アドレスを実サーバとして指定し、実サーバ コンフィギュレーション サブモードを開始します。
ステップ 10	`Switch-A(config-slb-real)# inservice`	ファイアウォールをイネーブルにします。
ステップ 11	`Switch-A(config-slb-real)# exit`	サーバ ファーム コンフィギュレーション モードに戻ります。
ステップ 12	`Switch-A(config-slb-sfarm)# real 10.0.102.200`	ファイアウォール 2 への経路上にある、CSM B のエイリアス IP アドレスを実サーバとして指定し、実サーバ コンフィギュレーション サブモードを開始します。
ステップ 13	`Switch-A(config-slb-real)# inservice`	ファイアウォールをイネーブルにします。

1. FORWARD-SF は実際には、実サーバ ファームではなく、トラフィックが（VLAN 10 経由で）インターネットに到達できるようにする、ルート フォワーディング ポリシーです。実サーバは含まれません。
2. このステップは、実サーバではなくフォワーディング ポリシーからなるサーバ ファームを設定する場合に必要です。
3. INSIDE-SF は、イントラネットから CSM B にトラフィックが到達できるようにする実サーバとして指定された、CSM B の 2 つのエイリアス IP アドレスからなります。
4. このステップは、ファイアウォールが含まれるサーバ ファームを設定する場合に必要です。
5. この作業は、サーバ ファームで保護されない側のファイアウォール インターフェイスを設定する場合に行ってください。

CSM A 上での仮想サーバの設定

CSM A 上で 3 つの仮想サーバを設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Switch-A(config)# module csm 5`	マルチモジュール コンフィギュレーション モードを開始し、CSM A がスロット 5 に搭載されていることを指定します。
ステップ 2	`Switch-A(config-module-csm)# vserver FORWARD-V101`	設定対象の仮想サーバとしてFORWARD-V101[1]を指定し、仮想サーバ コンフィギュレーション モードを開始します。
ステップ 3	`Switch-A(config-slb-vserver)# virtual 0.0.0.0 0.0.0.0 any`	あらゆるIPアドレスおよびあらゆるプロトコルと一致することを指定します[2]。

	コマンド	目的
ステップ 4	`Switch-A(config-slb-vserver))# vlan 101`	仮想サーバが VLAN 101 に届いたトラフィック、すなわちファイアウォールの保護されていない側からのトラフィックだけを受け付けることを指定します。
ステップ 5	`Switch-A(config-slb-vserver)# serverfarm FORWARD-SF`	この仮想サーバに対応するサーバ ファームを指定します[3]。
ステップ 6	`Switch-A(config-slb-vserver)# inservice`	仮想サーバをイネーブルにします。
ステップ 7	`Switch-A(config-slb-vserver)# exit`	マルチモジュール コンフィギュレーション モードに戻ります。
ステップ 8	`Switch-A(config-module-csm)# vserver FORWARD-V103`	設定対象の仮想サーバとしてFORWARD-V103[4]を指定し、仮想サーバ コンフィギュレーション モードを開始します。
ステップ 9	`Switch-A(config-slb-vserver)# virtual 0.0.0.0 0.0.0.0 any`	あらゆるIPアドレスおよびあらゆるプロトコルと一致することを指定します[5]。
ステップ 10	`Switch-A(config-slb-vserver))# vlan 103`	仮想サーバが VLAN 103 に届いたトラフィック、すなわちファイアウォールの保護されていない側からのトラフィックだけを受け付けることを指定します。
ステップ 11	`Switch-A(config-slb-vserver)# serverfarm FORWARD-SF`	この仮想サーバに対応するサーバ ファームを指定します[3]。
ステップ 12	`Switch-A(config-slb-vserver)# inservice`	仮想サーバをイネーブルにします。
ステップ 13	`Switch-A(config-slb-vserver)# exit`	マルチモジュール コンフィギュレーション モードに戻ります。
ステップ 14	`Switch-A(config-module-csm)# vserver OUTSIDE-VS`	設定対象の仮想サーバとして OUTSIDE-VS[6] を指定し、仮想サーバ コンフィギュレーション モードを開始します。
ステップ 15	`Switch-A(config-slb-vserver)# virtual 10.1.0.0 255.255.255.0 any`	この仮想サーバの IP アドレス、ネットマスク、およびプロトコル（あれば）を指定します。クライアントはこのアドレスによって、この仮想サーバが提供するサーバ ファームに到達します。
ステップ 16	`Switch-A(config-slb-vserver))# vlan 10`	仮想サーバが VLAN 10 に届いたトラフィック、すなわちインターネットからのトラフィックだけを受け付けることを指定します。
ステップ 17	`Switch-A(config-slb-vserver)# serverfarm TO-INSIDE-SF`	この仮想サーバに対応するサーバ ファームを指定します[7]。
ステップ 18	`Switch-A(config-slb-vserver)# inservice`	仮想サーバをイネーブルにします。

1. FORWARD-V101 は、インターネット トラフィックを（VLAN 101 経由で）ファイアウォールの保護されていない側に送ります。
2. クライアントの一致を制限するのは、VLAN 制約だけです（ステップ 4 を参照）。
3. このサーバ ファームは、実サーバからなる実サーバ ファームではなく、実際にはフォワーディング プレディクタです。
4. FORWARD-V103 は、インターネット トラフィックを（VLAN 103 経由で）ファイアウォールの保護されていない側に送ります。
5. クライアントの常に一致を制限するのは、VLAN 制約だけです（ステップ 10 を参照）。
6. OUTSIDE-VS は、インターネットからのトラフィックを（VLAN 10 経由で）CSM A に送ります。
7. サーバ ファームは、ファイアウォール 1 およびファイアウォール 2 の経路上にある、CSM B のエイリアス IP アドレスで構成されます。

CSM B の設定（ステルス ファイアウォールの例）

標準の設定例を作成するには、CSM B に対して次の設定作業が必要です。

- スイッチ B 上での VLAN の作成（p.11-13）
- CSM B 上での VLAN の設定（p.11-13）
- CSM B 上でのサーバ ファームの設定（p.11-14）
- CSM B 上での仮想サーバの設定（p.11-16）

（注）　設定作業は CSM A でも CSM B でも同じですが、手順、入力するコマンド、およびパラメータが異なります。

スイッチ B 上での VLAN の作成

スイッチ B 上で 3 つの VLAN を作成する手順は、次のとおりです。

（注）　この例では、CSM がそれぞれ別個の Catalyst 6500 シリーズ スイッチに搭載されているものとします。同一シャーシに搭載されている場合は、同じ Catalyst 6500 シリーズ スイッチのコンソールですべての VLAN を作成できます。

	コマンド	目的
ステップ 1	`Switch-B(config)# vlan`	VLAN モードを開始します[1]。
ステップ 2	`Switch-B(vlan)# vlan 102`	VLAN 102 を作成します[2]。
ステップ 3	`Switch-B(vlan)# vlan 104`	VLAN 104 を作成します[3]。
ステップ 4	`Switch-B(vlan)# vlan 200`	VLAN 200 を作成します[4]。

1. この作業は、CSM B が搭載されたスイッチのコンソールで行います。
2. VLAN 102 は、ファイアウォール 1 経由で CSM A に接続します。
3. VLAN 104 は、ファイアウォール 2 経由で CSM A に接続します。
4. VLAN 200 は、内部ネットワークに接続します。

CSM B 上での VLAN の設定

3 つの VLAN を設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Switch-B(config)# module csm 6`	マルチモジュール コンフィギュレーション モードを開始し、CSM B がスロット 6 に搭載されていることを指定します。
ステップ 2	`Switch-B(config-module-csm)# vlan 102 server`	設定対象の VLAN として VLAN 102 を指定し、サーバ VLAN であることを指定し、VLAN コンフィギュレーション モードを開始します。
ステップ 3	`Switch-B(config-slb-vlan-server)# ip address 10.0.101.36 255.255.255.0`	VLAN 102 の IP アドレスおよびネットマスクを指定します。
ステップ 4	`Switch-B(config-slb-vlan-server)# alias 10.0.101.200 255.255.255.0`	VLAN 102 用のエイリアス IP アドレスおよびネットマスクを指定します[1]。
ステップ 5	`Switch-B(config-slb-vlan-server)# exit`	マルチモジュール コンフィギュレーション モードに戻ります。

	コマンド	目的
ステップ 6	`Switch-B(config-module-csm)# vlan 104 server`	設定対象の VLAN として VLAN 104 を指定し、サーバ VLAN であることを指定し、VLAN コンフィギュレーション モードを開始します。
ステップ 7	`Switch-B(config-slb-vlan-server)# ip address 10.0.102.36 255.255.255.0`	VLAN 104 の IP アドレスおよびネットマスクを指定します。
ステップ 8	`Switch-B(config-slb-vlan)# alias 10.0.102.200 255.255.255.0`	VLAN 104 用のエイリアス IP アドレスおよびネットマスクを指定します[1]。
ステップ 9	`Switch-B(config-slb-vlan-server)# exit`	マルチモジュール コンフィギュレーション モードに戻ります。
ステップ 10	`Switch-B(config-module-csm)# vlan 20 server`	設定対象の VLAN として VLAN 20 を指定し、サーバ VLAN であることを指定し、VLAN コンフィギュレーション モードを開始します。
ステップ 11	`Switch-B(config-slb-vlan-server)# ip address 10.1.0.36 255.255.255.0`	VLAN 20 の IP アドレスおよびネットマスクを指定します。

1. このステップで、ロードバランシングの決定に使用する、CSM A のターゲットを特定します。

CSM B 上でのサーバ ファームの設定

CSM B 上で 3 つのサーバ ファームを設定する手順は、次のとおりです。

（注） SERVERS-SF では、この例ですでに **natpool** コマンドで作成した、クライアント NAT アドレス プールを使用して、クライアント NAT を実行することを指定します。コマンドを参照する前に、NAT プールを作成する必要があります。

	コマンド	目的
ステップ 1	`Switch-B(config)# module csm 6`	マルチモジュール コンフィギュレーション モードを開始し、CSM B がスロット 6 に搭載されていることを指定します。
ステップ 2	`Switch-B(config-module-csm)# serverfarm FORWARD-SF`	FORWARD-SF[1] サーバ ファーム（実際にはフォワーディング ポリシー）を作成して名前を指定し、サーバ ファーム コンフィギュレーション モードを開始します。
ステップ 3	`Switch-B(config-slb-sfarm)# no nat server`	サーバの IP アドレスおよびポート番号の NAT をディセーブルにします[2]。
ステップ 4	`Switch-B(config-slb-sfarm)# predictor forward`	ロードバランス アルゴリズムではなく、内部ルーティング テーブルに従って、トラフィックを転送します。
ステップ 5	`Switch-B(config-slb-sfarm)# exit`	マルチモジュール コンフィギュレーション モードに戻ります。
ステップ 6	`Switch-B(config-module-csm)# serverfarm TO-OUTSIDE-SF`	GENERIC-SF サーバ ファームを作成して名前を指定し、サーバ ファーム コンフィギュレーション モードを開始します[3]。
ステップ 7	`Switch-B(config-slb-sfarm)# no nat server`	サーバの IP アドレスおよびポート番号の NAT をディセーブルにします[4]。

	コマンド	目的
ステップ 8	`Switch-B(config-slb-sfarm)# real 10.0.101.100`	ファイアウォール 1 への経路上にある、CSM A のエイリアス IP アドレスを実サーバとして指定し、実サーバ コンフィギュレーション サブモードを開始します。
ステップ 9	`Switch-B(config-slb-real)# inservice`	実サーバ（実際にはエイリアス IP アドレス）をイネーブルにします。
ステップ 10	`Switch-B(config-slb-real)# exit`	サーバ ファーム コンフィギュレーション モードに戻ります。
ステップ 11	`Switch-B(config-slb-sfarm)# real 10.0.102.100`	ファイアウォール 2 への経路上にある、CSM B のエイリアス IP アドレスを実サーバとして指定し、実サーバ コンフィギュレーション サブモードを開始します。
ステップ 12	`Switch-B(config-slb-real)# inservice`	実サーバ（実際にはエイリアス IP アドレス）をイネーブルにします。
ステップ 13	`Switch-B(config-slb-real)# exit`	サーバ ファーム コンフィギュレーション モードに戻ります。
ステップ 14	`Switch-B(config-module-csm)# serverfarm SERVERS-SF`	SERVERS-SF[5] サーバ ファームを作成して名前を指定し、サーバ ファーム コンフィギュレーション モードを開始します。
ステップ 15	`Switch-B(config-slb-sfarm)# real 10.1.0.101`	イントラネット内のサーバを実サーバとして指定し、IP アドレスを割り当てて、実サーバ コンフィギュレーション サブモードを開始します。
ステップ 16	`Switch-B(config-slb-real)# inservice`	実サーバをイネーブルにします。
ステップ 17	`Switch-B(config-slb-real)# exit`	サーバ ファーム コンフィギュレーション モードに戻ります。
ステップ 18	`Switch-B(config-slb-sfarm)# real 10.1.0.102`	イントラネット内のサーバを実サーバとして指定し、IP アドレスを割り当てて、実サーバ コンフィギュレーション サブモードを開始します。
ステップ 19	`Switch-B(config-slb-real)# inservice`	実サーバをイネーブルにします。
ステップ 20	`Switch-B(config-slb-sfarm)# real 10.1.0.103`	イントラネット内のサーバを実サーバとして指定し、IP アドレスを割り当てて、実サーバ コンフィギュレーション サブモードを開始します。
ステップ 21	`Switch-B(config-slb-real)# inservice`	実サーバをイネーブルにします。

1. FORWARD-SF は実際には、実サーバ ファームではなく、トラフィックが（VLAN 20 経由で）イントラネットに到達できるようにする、ルート フォワーディング ポリシーです。実サーバは含まれません。
2. このステップは、実サーバではなくフォワーディング ポリシーからなるサーバ ファームを設定する場合に必要です。
3. OUTSIDE-SF は、イントラネットから CSM A にトラフィックが到達できるようにする実サーバとして指定された、CSM A の 2 つのエイリアス IP アドレスからなります。
4. このステップは、実サーバではなくフォワーディング ポリシーからなるサーバ ファームを設定する場合に必要です。
5. SERVERS-SF は、イントラネット内に配置された実サーバの IP アドレスからなります。

CSM B 上での仮想サーバの設定

CSM 上で 3 つの仮想サーバを設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Switch-B(config)# module csm 6`	マルチモジュール コンフィギュレーション モードを開始し、CSM B がスロット 6 に搭載されていることを指定します。
ステップ 2	`Switch-B(config-module-csm)# vserver FORWARD-VS-102`	設定対象の仮想サーバとして FORWARD-VS を指定し、仮想サーバ コンフィギュレーション モードを開始します。
ステップ 3	`Switch-B(config-slb-vserver)# virtual 0.0.0.0 0.0.0.0 any`	あらゆる IP アドレスおよびあらゆるプロトコルと一致することを指定します[1]。
ステップ 4	`Switch-B(config-slb-vserver)# vlan 102`	仮想サーバが VLAN 102 に届いたトラフィック、すなわちファイアウォール 1 の保護されている側からのトラフィックだけを受け付けることを指定します。
ステップ 5	`Switch-B(config-slb-vserver)# serverfarm FORWARD-SF`	この仮想サーバに対応するサーバ ファームを指定します[2]。
ステップ 6	`Switch-B(config-slb-vserver)# inservice`	仮想サーバをイネーブルにします。
ステップ 7	`Switch-B(config-slb-vserver)# exit`	マルチモジュール コンフィギュレーション モードに戻ります。
ステップ 8	`Switch-B(config-module-csm)# vserver FORWARD-VS-104`	設定対象の仮想サーバとして FORWARD-VS[3] を指定し、仮想サーバ コンフィギュレーション モードを開始します。
ステップ 9	`Switch-B(config-slb-vserver)# virtual 0.0.0.0 0.0.0.0 any`	あらゆる IP アドレスおよびあらゆるプロトコルと一致することを指定します[1]。
ステップ 10	`Switch-B(config-slb-vserver)# vlan 104`	仮想サーバが VLAN 104 に届いたトラフィック、すなわちファイアウォール 2 の保護されている側からのトラフィックだけを受け付けることを指定します。
ステップ 11	`Switch-B(config-slb-vserver)# serverfarm FORWARD-SF`	この仮想サーバに対応するサーバ ファームを指定します[2]。
ステップ 12	`Switch-B(config-slb-vserver)# inservice`	仮想サーバをイネーブルにします。
ステップ 13	`Switch-B(config-slb-vserver)# exit`	マルチモジュール コンフィギュレーション モードに戻ります。
ステップ 14	`Switch-B(config-module-csm)# vserver INSIDE-VS`	設定対象の仮想サーバとして INSIDE-VS[4] を指定し、仮想サーバ コンフィギュレーション モードを開始します。
ステップ 15	`Switch-B(config-slb-vserver)# virtual 0.0.0.0 0.0.0.0 any`	あらゆる IP アドレスおよびあらゆるプロトコルと一致することを指定します[1]。
ステップ 16	`Switch-B(config-slb-vserver)# vlan 20`	仮想サーバが VLAN 20 に届いたトラフィック、すなわちイントラネットからのトラフィックだけを受け付けることを指定します。
ステップ 17	`Switch-B(config-slb-vserver)# serverfarm TO-OUTSIDE-SF`	この仮想サーバに対応するサーバ ファーム（実サーバとしてのCSM Aのエイリアス IP アドレスからなり、トラフィックをファイアウォール 1 および 2 に流す）を指定し、実サーバ コンフィギュレーション サブモードを開始します。

	コマンド	目的
ステップ 18	`Switch-B(config-slb-vserver)#` **`inservice`**	仮想サーバをイネーブルにします。
ステップ 19	`Switch-B(config-slb-vserver)#` **`exit`**	マルチモジュール コンフィギュレーション モードに戻ります。
ステップ 20	`Switch-B(config-module-csm)#` **`vserver TELNET-VS`**	設定対象の仮想サーバとして TELNET-VS[5] を指定し、仮想サーバ コンフィギュレーション モードを開始します。 **（注）** TELNET-VS は VLAN 制限を使用しません。したがって、（ファイアウォールまたは内部ネットワークからの）あらゆる送信元のトラフィックがこのアドレス経由で負荷分散されます。
ステップ 21	`Switch-B(config-slb-vserver)#` **`virtual 10.1.0.200 255.255.255.0 tcp telnet`**	この仮想サーバの IP アドレス、ネットマスク、プロトコル（TCP）、およびポート（Telnet）を指定します[6]。
ステップ 22	`Switch-B(config-slb-vserver)#` **`serverfarm SERVERS-SF`**	この仮想サーバに対応する、実サーバからなるサーバファームを指定します。
ステップ 23	`Switch-B(config-slb-vserver)#` **`inservice`**	仮想サーバをイネーブルにします。

1. クライアントの一致を制限するのは、VLAN 制約だけです。
2. このサーバ ファームは、実サーバからなる実サーバ ファームではなく、実際にはフォワーディング プレディクタです。
3. FORWARD-VS は、インターネットからのトラフィックを（VLAN 20 経由で）イントラネットに送ります。
4. INSIDE-VS は、イントラネットからのトラフィックをファイアウォール 1 経由（VLAN 102 および 101 経由）またはファイアウォール 2 経由（VLAN 104 および 103 経由）で CSM A に送ります。
5. TELNET-VS は、インターネットからのトラフィックを内部ネットワーク内の Telnet サーバに送ります。
6. クライアントはこのアドレスによって、この仮想サーバが提供するサーバ ファームに到達します。

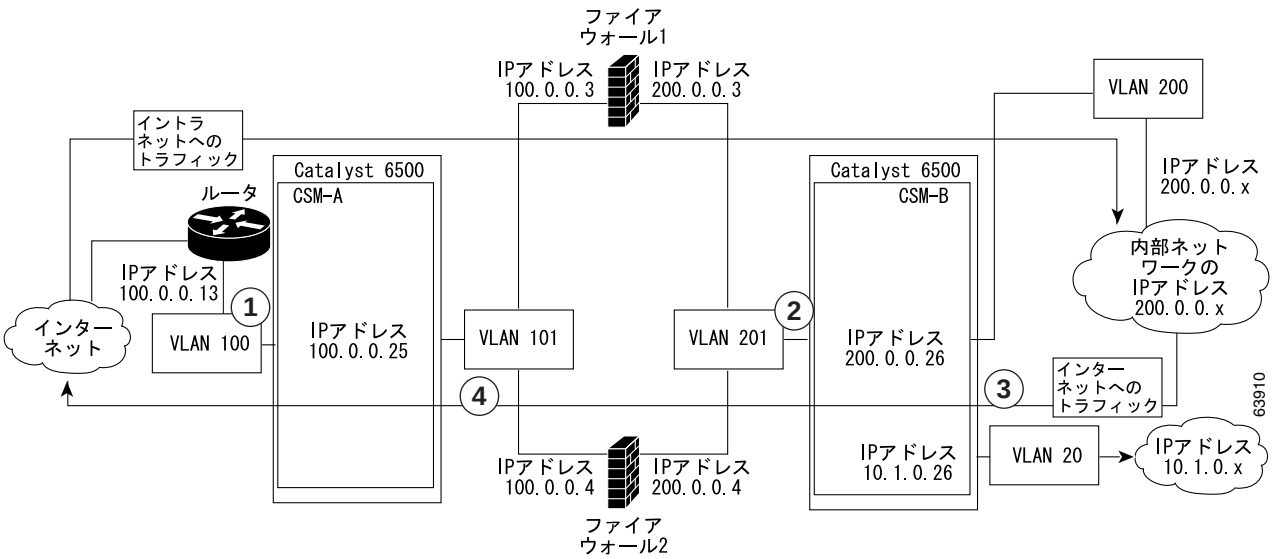
標準ファイアウォール ロードバランシングの設定

ここでは、標準ファイアウォール用にファイアウォール ロードバランシングを設定する方法について説明します。

標準ファイアウォール構成の場合のパケット フロー

標準ファイアウォール設定では、ファイアウォールは 2 つの異なる VLAN に接続し、接続先 VLAN の IP アドレスを指定して設定します（図 11-7 を参照）。

図 11-7　標準ファイアウォールの設定例



アイテム	トラフィックの方向	入口	出口
1	イントラネットへ	VLAN 100	VLAN 101
2	イントラネットへ	VLAN 201	VLAN 200 および 20
3	インターネットへ	VLAN 200 および 20	VLAN 201
4	インターネットへ	VLAN 101	VLAN 100

図 11-7 では、2 つの標準ファイアウォール（ファイアウォール 1 およびファイアウォール 2）が 2 つの CSM（CSM A および CSM B）の間にあります。トラフィックは共有 VLAN（VLAN 101 および VLAN 201）を介してファイアウォールを出入りします。どちらの標準ファイアウォールも、各共有 VLAN 上に固有のアドレスを持っています。

VLAN はインターネット（VLAN 100）、内部ネットワーク（VLAN 200）、および内部サーバ ファーム（VLAN 20）に接続できるようにします。

CSM は、実サーバの場合と同様、標準ファイアウォール間でトラフィックを分散させます。標準ファイアウォールは、実サーバと同様、IP アドレスを指定してサーバ ファーム内で設定します。標準ファイアウォールが所属するサーバ ファームは、ロードバランス プレディクタが割り当てられ、仮想サーバと関連付けられます。

標準ファイアウォールの設定例

標準ファイアウォールの設定例では、2 つの CSM（CSM A および CSM B）をそれぞれ別個の Catalyst 6500 シリーズ スイッチに搭載しています。

（注）　この例を使用できるのは、同じ Catalyst 6500 シリーズ スイッチ シャーシに搭載された 2 つのモジュールを設定する場合です。また、CSM A および CSM B の両方を設定するときに、その CSM のスロット番号を指定することによって、単一スイッチ シャーシに 1 つだけ搭載された CSM を設定する場合にも、この例を使用できます。

CSM A の設定（標準ファイアウォールの例）

標準の設定例を作成するには、CSM A に対して次の設定作業が必要です。

- スイッチ A 上での VLAN の作成（p.11-19）
- CSM A 上での VLAN の設定（p.11-20）
- CSM A 上でのサーバ ファームの設定（p.11-20）
- CSM A 上での仮想サーバの設定（p.11-21）

（注）　設定作業は CSM A でも CSM B でも同じですが、手順、入力するコマンド、およびパラメータが異なります。

スイッチ A 上での VLAN の作成

図 11-7 に示した例では、スイッチ A 上で VLAN を 2 つ作成する必要があります。

（注）　この例では、CSM がそれぞれ別個の Catalyst 6500 シリーズ スイッチ シャーシに搭載されているものとします。同一シャーシに搭載されている場合は、同じ Catalyst 6500 シリーズ スイッチのコンソールですべての VLAN を作成できます。

スイッチ A 上で VLAN を作成する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Switch-A(config)# vlan`	VLAN モードを開始します[1]。
ステップ 2	`Switch-A(vlan)# vlan 100`	VLAN 100 を作成します[2]。
ステップ 3	`Switch-A(vlan)# vlan 101`	VLAN 101 を作成します[3]。

1. この作業は、CSM A が搭載されたスイッチのコンソールで行います。
2. VLAN 100 は CSM A をインターネットに接続します。
3. VLAN 101 は CSM A をファイアウォールの保護されていない側に接続します。

CSM A 上での VLAN の設定

2 つの VLAN を設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Switch-A(config)# module csm 5`	マルチモジュール コンフィギュレーション モードを開始し、CSM A がスロット 5 に搭載されていることを指定します。
ステップ 2	`Switch-A(config-module-csm)# vlan 100 client`	設定対象の VLAN として VLAN 100 を指定し、クライアント VLAN であることを指定し、VLAN コンフィギュレーション モードを開始します。
ステップ 3	`Switch-A(config-slb-vlan-client)# ip address 100.0.0.25 255.255.255.0`	VLAN 100 の IP アドレスおよびネットマスクを指定します。
ステップ 4	`Switch-A(config-slb-vlan-client)# gateway 100.0.0.13`	CSM A のインターネット側ルータのゲートウェイ IP アドレスを設定します。
ステップ 5	`Switch-A(config-slb-vlan-client)# exit`	マルチモジュール コンフィギュレーション モードに戻ります。
ステップ 6	`Switch-A(config-module-csm)# vlan 101 server`	設定対象の VLAN として VLAN 101 を指定し、サーバ VLAN であることを指定し、VLAN コンフィギュレーション モードを開始します。
ステップ 7	`Switch-A(config-slb-vlan-server)# ip address 100.0.0.25 255.255.255.0`	VLAN 101 の IP アドレスおよびネットマスクを指定します。
ステップ 8	`Switch-A(config-slb-vlan-server)# alias 100.0.0.20 255.255.255.0`	VLAN 101 用のエイリアス IP アドレスおよびネットマスクを指定します[1]。

1. このステップで、ロードバランシングの決定に使用する、CSM B のターゲットを特定します。

CSM A 上でのサーバ ファームの設定

（注） ファイアウォール 1 およびファイアウォール 2 の保護された側の IP アドレスは、CSM B と関連付けられた SEC-SF サーバ ファーム内の実サーバとして設定します。

CSM A 上で 2 つのサーバ ファームを設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Switch-A(config)# module csm 5`	マルチモジュール コンフィギュレーション モードを開始し、CSM A がスロット 5 に搭載されていることを指定します。
ステップ 2	`Switch-A(config-module-csm)# serverfarm FORWARD-SF`	FORWARD-SF[1] サーバ ファーム（実際にはフォワーディング ポリシー）を作成して名前を指定し、サーバ ファーム コンフィギュレーション モードを開始します。
ステップ 3	`Switch-A(config-slb-sfarm)# no nat server`	サーバの IP アドレスおよびポート番号の NAT をディセーブルにします[2]。
ステップ 4	`Switch-A(config-slb-sfarm)# predictor forward`	ロードバランス アルゴリズムではなく、内部ルーティング テーブルに従って、トラフィックを転送します。

	コマンド	目的
ステップ 5	`Switch-A(config-slb-sfarm)# exit`	マルチモジュール コンフィギュレーション モードに戻ります。
ステップ 6	`Switch-A(config-module-csm)# serverfarm INSEC-SF`	（実サーバとしてのファイアウォールが含まれる）INSEC-SF[3] サーバ ファームを作成して名前を指定し、サーバ ファーム コンフィギュレーション モードを開始します。
ステップ 7	`Switch-A(config-slb-sfarm)# no nat server`	サーバの IP アドレスおよびポート番号の NAT をディセーブルにします[4]。
ステップ 8	`Switch-A(config-slb-sfarm)# predictor hash address source 255.255.255.255`	送信元 IP アドレスに基づくハッシュ値を使用して、サーバを選択します[5]。
ステップ 9	`Switch-A(config-slb-sfarm)# real 100.0.0.3`	ファイアウォール 1 を実サーバとして設定し、ファイアウォールの保護されない側に IP アドレスを割り当て、実サーバ コンフィギュレーション サブモードを開始します。
ステップ 10	`Switch-A(config-slb-real)# inservice`	ファイアウォールをイネーブルにします。
ステップ 11	`Switch-A(config-slb-real)# exit`	サーバ ファーム コンフィギュレーション モードに戻ります。
ステップ 12	`Switch-A(config-slb-sfarm)# real 100.0.0.4`	ファイアウォール 2 を実サーバとして設定し、ファイアウォールの保護されない側に IP アドレスを割り当て、実サーバ コンフィギュレーション サブモードを開始します。
ステップ 13	`Switch-A(config-slb-real)# inservice`	ファイアウォールをイネーブルにします。

1. FORWARD-SF は実際には、実サーバ ファームではなく、トラフィックが（VLAN 100 経由で）インターネットに到達できるようにする、ルート フォワーディング ポリシーです。実サーバは含まれません。
2. このステップは、実サーバではなくフォワーディング ポリシーからなるサーバ ファームを設定する場合に必要です。
3. INSEC-SF にはファイアウォール 1 およびファイアウォール 2 が含まれます。それぞれの保護されていない側の IP アドレスをこのサーバ ファーム内の実サーバとして設定します。
4. このステップは、ファイアウォールが含まれるサーバ ファームを設定する場合に必要です。
5. この作業は、サーバ ファームで保護されない側のファイアウォール インターフェイスを設定する場合に行ってください。

CSM A 上での仮想サーバの設定

CSM A 上で 2 つの仮想サーバを設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Switch-A(config)# module csm 5`	マルチモジュール コンフィギュレーション モードを開始し、CSM A がスロット 5 に搭載されていることを指定します。
ステップ 2	`Switch-A(config-module-csm)# vserver FORWARD-VS`	設定対象の仮想サーバとして FORWARD-VS[1] を指定し、仮想サーバ コンフィギュレーション モードを開始します。
ステップ 3	`Switch-A(config-slb-vserver)# virtual 0.0.0.0 0.0.0.0 any`	あらゆるIPアドレスおよびあらゆるプロトコルと一致することを指定します[2]。
ステップ 4	`Switch-A(config-slb-vserver))# vlan 101`	仮想サーバが VLAN 101 に届いたトラフィック、すなわちファイアウォールの保護されていない側からのトラフィックだけを受け付けることを指定します。

	コマンド	目的
ステップ 5	`Switch-A(config-slb-vserver)# serverfarm FORWARD-SF`	この仮想サーバに対応するサーバ ファームを指定します[3]。
ステップ 6	`Switch-A(config-slb-vserver)# inservice`	仮想サーバをイネーブルにします。
ステップ 7	`Switch-A(config-slb-vserver)# exit`	マルチモジュール コンフィギュレーション モードに戻ります。
ステップ 8	`Switch-A(config-module-csm)# vserver INSEC-VS`	設定対象の仮想サーバとして INSEC-VS[4] を指定し、仮想サーバ コンフィギュレーション モードを開始します。
ステップ 9	`Switch-A(config-slb-vserver)# virtual 200.0.0.0 255.255.255.0 any`	この仮想サーバの IP アドレス、ネットマスク、およびプロトコル（あれば）を指定します[5]。
ステップ 10	`Switch-A(config-slb-vserver))# vlan 100`	仮想サーバが VLAN 100 に届いたトラフィック、すなわちインターネットからのトラフィックだけを受け付けることを指定します。
ステップ 11	`Switch-A(config-slb-vserver)# serverfarm INSEC-SF`	この仮想サーバに対応するサーバ ファームを指定します[6]。
ステップ 12	`Switch-A(config-slb-vserver)# inservice`	仮想サーバをイネーブルにします。

1. FORWARD-VS は、インターネット トラフィックを（VLAN 101 経由で）ファイアウォールの保護されていない側に送ります。
2. クライアントの一致を制限するのは、VLAN 制約だけです（ステップ 4 を参照）。
3. このサーバ ファームは、実サーバからなる実サーバ ファームではなく、実際にはフォワーディング プレディクタです。
4. INSEC-VS は、インターネットからのトラフィックを（VLAN 101 経由で）CSM A に送ります。
5. クライアントはこのアドレスによって、この仮想サーバが提供するサーバ ファームに到達します。
6. サーバ ファームは実サーバではなくファイアウォールからなります。

CSM B の設定（標準ファイアウォールの例）

標準の設定例を作成するには、CSM B に対して次の設定作業が必要です。

- スイッチ B 上での VLAN の作成（p.11-22）
- CSM B 上での VLAN の設定（p.11-23）
- CSM B 上でのサーバ ファームの設定（p.11-24）
- CSM B 上での仮想サーバの設定（p.11-25）

（注） 設定作業は CSM A でも CSM B でも同じですが、手順、入力するコマンド、およびパラメータが異なります。

スイッチ B 上での VLAN の作成

（注） この例では、CSM がそれぞれ別個の Catalyst 6500 シリーズ スイッチ シャーシに搭載されているものとします。同一シャーシに搭載されている場合は、同じ Catalyst 6500 シリーズ スイッチのコンソールですべての VLAN を作成できます。

スイッチ B 上で 3 つの VLAN を作成する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Switch-B(config)# vlan`	VLAN モードを開始します[1]。
ステップ 2	`Switch-B(vlan)# vlan 201`	VLAN 201 を作成します[2]。
ステップ 3	`Switch-B(vlan)# vlan 200`	VLAN 200 を作成します[3]。
ステップ 4	`Switch-B(vlan)# vlan 20`	VLAN 20 を作成します[4]。

1. この作業は、CSM B が搭載されたスイッチのコンソールで行います。
2. VLAN 201 はファイアウォールの保護されている側に接続します。
3. VLAN 200 は、内部ネットワークに接続します。
4. VLAN 20 は、内部サーバ ファームに接続します。

CSM B 上での VLAN の設定

CSM B 上で 3 つの VLAN を設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Switch-B(config)# module csm 6`	マルチモジュール コンフィギュレーション モードを開始し、CSM B がスロット 6 に搭載されていることを指定します。
ステップ 2	`Switch-B(config-module-csm)# vlan 201 server`	設定対象の VLAN として VLAN 201 を指定し、サーバ VLAN であることを指定し、VLAN コンフィギュレーション モードを開始します。
ステップ 3	`Switch-B(config-slb-vlan-server)# ip address 200.0.0.26 255.255.255.0`	VLAN 201 の IP アドレスおよびネットマスクを指定します。
ステップ 4	`Switch-B(config-slb-vlan-server)# alias 200.0.0.20 255.255.255.0`	VLAN 201 用のエイリアス IP アドレスおよびネットマスクを指定します[1]。
ステップ 5	`Switch-B(config-slb-vlan-server)# exit`	VLAN コンフィギュレーション モードに戻ります。
ステップ 6	`Switch-B(config-module-csm)# vlan 20 server`	設定対象の VLAN として VLAN 20 を指定し、サーバ VLAN であることを指定し、VLAN コンフィギュレーション モードを開始します。
ステップ 7	`Switch-B(config-slb-vlan-server)# ip address 10.1.0.26 255.255.255.0`	VLAN 20 の IP アドレスおよびネットマスクを指定します。
ステップ 8	`Switch-B(config-slb-vlan-server)# exit`	VLAN コンフィギュレーション モードに戻ります。
ステップ 9	`Switch-B(config-module-csm)# vlan 200 client`	設定対象の VLAN として VLAN 200 を指定し、クライアント VLAN であることを指定し、VLAN コンフィギュレーション モードを開始します。
ステップ 10	`Switch-B(config-slb-vlan)# ip address 200.0.0.26 255.255.255.0`	VLAN 200 の IP アドレスおよびネットマスクを指定します。

1. このステップで、ロードバランシングの決定に使用する、CSM A のターゲットを特定します。

CSM B 上でのサーバ ファームの設定

（注）　ファイアウォール 1 およびファイアウォール 2 の保護された側の IP アドレスは、CSM A と関連付けられた INSEC-SF サーバ ファーム内の実サーバとして設定します。

CSM B 上で 2 つのサーバ ファームを設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Switch-B(config)# module csm 6`	マルチモジュール コンフィギュレーション モードを開始し、CSM B がスロット 6 に搭載されていることを指定します。
ステップ 2	`Switch-B(config-module-csm)# serverfarm GENERIC-SF`	GENERIC-SF[1] サーバ ファームを作成して名前を指定し、サーバ ファーム コンフィギュレーション モードを開始します。
ステップ 3	`Switch-B(config-slb-sfarm)# real 10.1.0.101`	内部サーバ ファームのサーバを実サーバとして指定し、IP アドレスを割り当てて、実サーバ コンフィギュレーション サブモードを開始します。
ステップ 4	`Switch-B(config-slb-real)# inservice`	実サーバをイネーブルにします。
ステップ 5	`Switch-B(config-slb-real)# exit`	サーバ ファーム コンフィギュレーション モードに戻ります。
ステップ 6	`Switch-B(config-slb-sfarm)# real 10.1.0.102`	内部サーバ ファームのサーバを実サーバとして指定し、IP アドレスを割り当てて、実サーバ コンフィギュレーション サブモードを開始します。
ステップ 7	`Switch-B(config-slb-real)# inservice`	実サーバをイネーブルにします。
ステップ 8	`Switch-B(config-slb-real)# exit`	サーバ ファーム コンフィギュレーション モードに戻ります。
ステップ 9	`Switch-B(config-slb-sfarm)# exit`	マルチモジュール コンフィギュレーション モードに戻ります。
ステップ 10	`Switch-B(config-module-csm)# serverfarm SEC-SF`	SEC-SF[2] サーバ ファームを作成して名前を指定し、サーバ ファーム コンフィギュレーション モードを開始します。
ステップ 11	`Switch-B(config-slb-sfarm)# no nat server`	サーバのIPアドレスおよびポート番号のNATをディセーブルにします[3]。
ステップ 12	`Switch-B(config-slb-sfarm)# predictor hash address destination 255.255.255.255`	宛先 IP アドレスに基づくハッシュ値を使用して、サーバを選択します[4]。
ステップ 13	`Switch-B(config-slb-sfarm)# real 200.0.0.3`	ファイアウォール 1 を実サーバとして設定し、ファイアウォールの保護されない側にIPアドレスを割り当て、実サーバ コンフィギュレーション サブモードを開始します。
ステップ 14	`Switch-B(config-slb-real)# inservice`	ファイアウォールをイネーブルにします。
ステップ 15	`Switch-B(config-slb-real)# exit`	サーバ ファーム コンフィギュレーション モードに戻ります。

	コマンド	目的
ステップ 16	`Switch-B(config-slb-sfarm)# real 200.0.0.4`	ファイアウォール 2 を実サーバとして設定し、ファイアウォールの保護されない側にIPアドレスを割り当て、実サーバ コンフィギュレーション サブモードを開始します。
ステップ 17	`Switch-B(config-slb-real)# inservice`	ファイアウォールをイネーブルにします。

1. GENERIC-SF は、内部サーバ ファーム内の実サーバからなります。
2. SEC-SF にはファイアウォール 1 およびファイアウォール 2 が含まれます。それぞれの保護される側の IP アドレスをこのサーバ ファーム内の実サーバとして設定します。
3. このステップは、ファイアウォールが含まれるサーバ ファームを設定する場合に必要です。
4. この作業は、サーバ ファームで保護されない側のファイアウォール インターフェイスを設定する場合に行ってください。

CSM B 上での仮想サーバの設定

CSM B 上で 3 つの仮想サーバを設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	`Switch-B(config)# module csm 6`	マルチモジュール コンフィギュレーション モードを開始し、CSM B がスロット 6 に搭載されていることを指定します。
ステップ 2	`Switch-B(config-module-csm)# vserver GENERIC-VS`	設定対象の仮想サーバとして GENERIC-VS[1] を指定し、仮想サーバ コンフィギュレーション モードを開始します。
ステップ 3	`Switch-B(config-slb-vserver)# virtual 200.0.0.127 tcp 0`	この仮想サーバの IP アドレス、プロトコル（TCP）、およびポート（0=any）を指定します[2]。
ステップ 4	`Switch-B(config-slb-vserver))# vlan 201`	仮想サーバが VLAN 201 に届いたトラフィック、すなわちファイアウォールの保護されている側からのトラフィックだけを受け付けることを指定します。
ステップ 5	`Switch-B(config-slb-vserver)# serverfarm GENERIC-SF`	この仮想サーバに対応するサーバ ファームを指定します[3]。
ステップ 6	`Switch-B(config-slb-vserver)# inservice`	仮想サーバをイネーブルにします。
ステップ 7	`Switch-B(config-slb-vserver)# exit`	マルチモジュール コンフィギュレーション モードに戻ります。
ステップ 8	`Switch-B(config-module-csm)# vserver SEC-20-VS`	設定対象の仮想サーバとして SEC-20-VS[4] を指定し、仮想サーバ コンフィギュレーション モードを開始します。
ステップ 9	`Switch-B(config-slb-vserver)# virtual 200.0.0.0 255.255.255.0 any`	この仮想サーバの IP アドレス、ネットマスク、およびプロトコル（あれば）を指定します[2]。
ステップ 10	`Switch-B(config-slb-vserver))# vlan 20`	仮想サーバが VLAN 20 に届いたトラフィック、すなわち内部サーバ ファームからのトラフィックだけを受け付けることを指定します。
ステップ 11	`Switch-B(config-slb-vserver)# serverfarm SEC-SF`	この仮想サーバに対応するサーバ ファームを指定します[5]。
ステップ 12	`Switch-B(config-slb-vserver)# inservice`	仮想サーバをイネーブルにします。
ステップ 13	`Switch-B(config-slb-vserver)# exit`	マルチモジュール コンフィギュレーション モードに戻ります。
ステップ 14	`Switch-B(config-module-csm)# vserver SEC-200-VS`	設定対象の仮想サーバとして SEC-20-VS[6] を指定し、仮想サーバ コンフィギュレーション モードを開始します。

	コマンド	目的
ステップ 15	`Switch-B(config-slb-vserver)#` **`virtual 200.0.0.0 255.255.255.0 any`**	この仮想サーバの IP アドレス、ネットマスク、およびプロトコル（あれば）を指定します[2]。
ステップ 16	`Switch-B(config-slb-vserver))#` **`vlan 200`**	仮想サーバが VLAN 200 に届いたトラフィック、すなわち内部ネットワークからのトラフィックだけを受け付けることを指定します。
ステップ 17	`Switch-B(config-slb-vserver)#` **`serverfarm SEC-SF`**	この仮想サーバに対応するサーバ ファームを指定します[5]。
ステップ 18	`Switch-B(config-slb-vserver)#` **`inservice`**	仮想サーバをイネーブルにします。

1. GENERIC-VS によって、インターネットを宛先とする、内部サーバ ファームおよび内部ネットワークからのトラフィックが、ファイアウォールの保護されている側に（VLAN 101 経由で）送られます。
2. クライアントはこのアドレスによって、この仮想サーバが提供するサーバ ファームに到達します。
3. サーバ ファームは、内部サーバ ファーム ネットワーク内にあります。
4. SEC-20-VS は、インターネットからのトラフィックを（VLAN 20 経由で）内部サーバ ファームに送ります。
5. サーバ ファームは実サーバではなくファイアウォールからなります。
6. SEC-200-VS は、インターネットからのトラフィックを（VLAN 20 経由で）内部ネットワークに送ります。

ファイアウォール用リバーススティッキの設定

リバーススティッキ機能では、クライアント IP アドレスに基づいたロードバランスの決定に関するデータベースを作成します。この機能によって、データベースにリバーススティッキ エントリがあった場合に、ロードバランスの決定が変更されます。データベースにリバーススティッキ エントリがなかった場合は、ロードバランスの決定が実行され、将来のマッチングのために結果が保存されます。

ファイアウォール用リバーススティッキの概要

リバーススティッキは、接続を反対方向からのものとみなして、スティッキ データベースにエントリを追加する 1 つの方法を提供します。リバーススティッキが行われた仮想サーバは、着信実サーバが含まれている指定のデータベースにエントリを追加します。

（注） 着信実サーバは、サーバ ファーム内の実サーバでなければなりません。

このエントリは、別の仮想サーバ上の sticky コマンドによってマッチングされます。他方の仮想サーバは、前もって作成されたこのエントリに基づいて、クライアントにトラフィックを送ります。

CSM は、送信元 IP キーから実サーバへのリンクとして、リバーススティッキ情報を保存します。ロードバランサがスティッキ データベースの割り当てられた仮想サーバと新しくセッションを開始するときには、最初にデータベースにエントリがすでにあるかどうかを確認します。一致するエントリがあった場合、セッションは指定された実サーバに接続されます。それ以外の場合は、スティッキ キーと適切な実サーバを結びつける、新しいエントリが作成されます。図 11-8 に、ファイアウォールでリバーススティッキ機能をどのように使用するかを示します。

図 11-8　ファイアウォール用リバーススティッキ

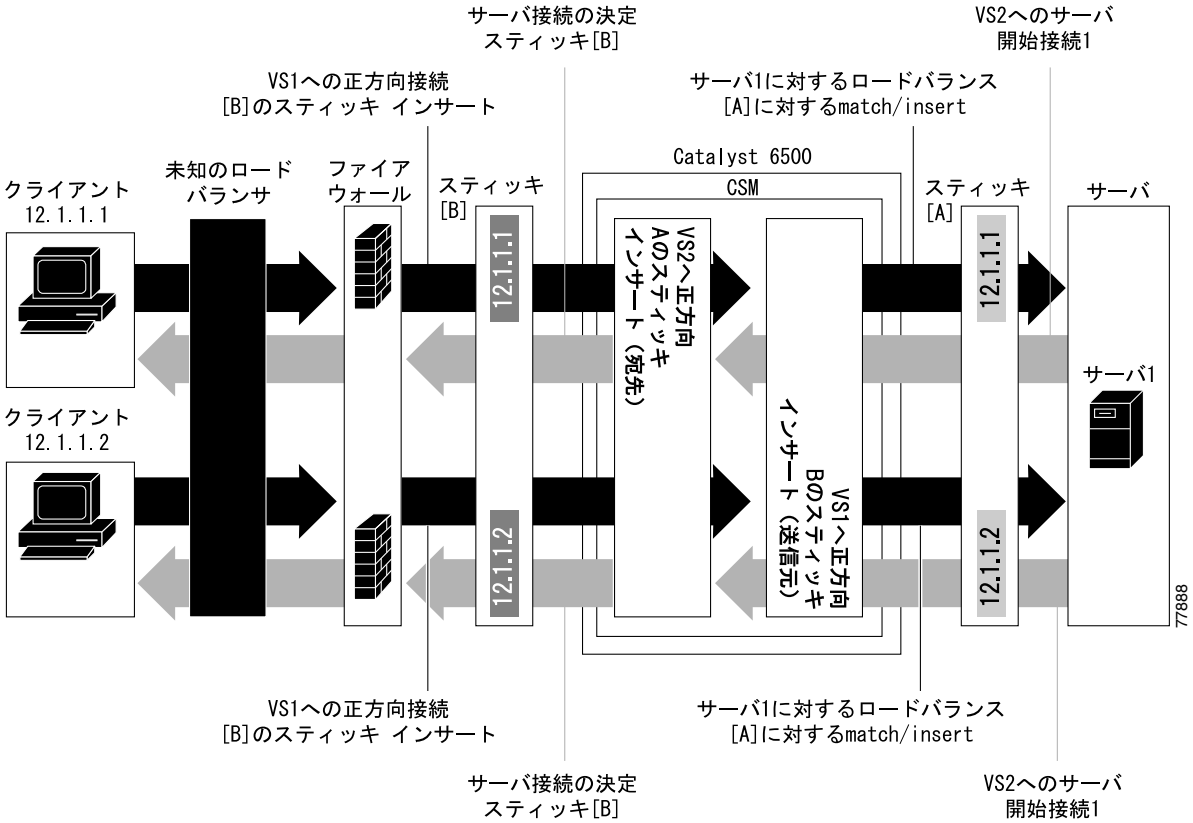


図 11-8 のリバーススティッキ プロセスは、次のとおりです。

- クライアントは、ロードバランス対象のファイアウォールを通過して、CSM 仮想サーバである VS1 に接続します。このロードバランスの決定は、CSM と対話しないで行われます。
- サーバ 1 は最初のクライアントに戻る接続を作成します。この接続は仮想サーバ VS2 と対応します。VS2 は、最初の VS1 リバーススティッキによって追加されたスティッキ情報を使用します。したがって、同じファイアウォール 1 に強制的に接続されます。
- 別のファイアウォールを通過する第 2 のクライアントは、同じ VS1 が接続します。リバーススティッキによって、第 2 のクライアント用にファイアウォール 2 を示す新しいエントリがデータベース B に作成されます。VS1 もサーバ 1 に対して通常のスティッキを実行します。
- サーバ 1 はクライアント 2 に戻る接続を作成します。この接続は VS2 の接続と一致します。VS2 は、最初の VS1 リバーススティッキによって追加されたスティッキ情報を使用します。この接続は、ファイアウォール 2 への接続に使用されます。
- サーバが最初の接続を開始すると、サーバに戻るリンクが VS2 によって作成され、通常のロードバランス決定によって一方のファイアウォールへの接続が作成されます。

（注） この設定では、任意のバランシング　メトリックを使用する正方向の接続（クライアントからサーバ）がサポートされます。ただし、サーバが開始したトラフィックへのクライアント応答が適切なファイアウォールに送られるようにするには、VS2 からファイアウォールへのバランシング メトリックが未知のロードバランサのメトリックと一致しなければなりません。または、未知のロードバランサが同様に新しい buddy 接続を固定（stick）しなければなりません。

ファイアウォール用リバーススティッキの設定

ファイアウォール ロードバランスのために IP リバーススティッキを設定する手順は、次のとおりです。

	コマンド	目的
ステップ 1	SLB-Switch(config)# **module csm** *slot*	特定の CSM モジュールにロードバランス コマンドを関連付け、指定したスロットに対して CSM モジュール コンフィギュレーション サブモードを開始します。
ステップ 2	SLB-Switch(config-module-csm)# **vserver** *virtserver-name*	仮想サーバを特定し、仮想サーバ コンフィギュレーション サブモードを開始します。
ステップ 3	SLB-Switch(config-slb-vserver)# **sticky** *duration* [**group** *group-id*] [**netmask** *ip-netmask*] [**source** | **destination** | **both**]	スティッキ エントリ キーに使用する IP 情報の部分（送信元、宛先、または両方）を定義します。
ステップ 4	SLB-Switch(config-slb-vserver)# **reverse-sticky** *group-id*	最初の送信元に戻る反対方向で、CSM が接続を維持するようにします。
ステップ 5	SLB-Switch# **show module csm** *slot* **sticky**	スティッキ データベースを表示します。

ステートフル ファイアウォール接続のリマッピングの設定

ファイアウォールの再割り当て機能を設定するには、Cisco IOS ソフトウェアの Release 12.1(19)E の MSFC イメージが必要です。

ファイアウォールの再割り当てを設定する手順は、次のとおりです。

ステップ 1　ファイアウォール用のサーバファーム サブモードで、次の動作を設定します。

```
Cat6k-2(config)# serverfarm FW-FARM
failaction reassign
```

ステップ 2　実サーバが失敗した場合（プローブまたは Address Resolution Protocol[ARP]）は、各ファイアウォール用のバックアップ実サーバを割り当てます。

```
Cat6k-2(config-slb-sfarm)# serverfarm FW-FARM
Cat6k-2(config-slb-sfarm)# real 1.1.1.1
Cat6k(config-slb-module-real)# backup real 2.2.2.2
Cat6k(config-slb-module-real)# inservice
Cat6k-2(config-slb-sfarm)# real 2.2.2.2
Cat6k(config-slb-module-real)# backup real 3.3.3.3
Cat6k(config-slb-module-real)# inservice
Cat6k-2(config-slb-sfarm)# real 3.3.3.3
Cat6k(config-slb-module-real)# backup real 1.1.1.1
Cat6k(config-slb-module-real)# inservice
```

ステップ 3　このサーバファーム用の Internet Control Message Protocol（ICMP）プローブ（ファイアウォールを経由）を設定します。

ステップ 4　ファイアウォールの外側および内側に CSM 用 ICMP プローブを設定します。

バックアップ実サーバが、同じ順序で CSM の両側に設定されていることを確認します。

接続の宛先または負荷分散先が失敗したプライマリ サーバの場合、実サーバに割り当てられた稼働中のスタンバイ オプションにより、このサーバが接続のみを受信するよう指定されます。real 2.2.2.2 として指定された実サーバを稼働中のスタンバイで設定する場合、すべての接続は real 1.1.1.1 または real 3.3.3.3 として指定された実サーバのいずれかに達します。実サーバ real 1.1.1.1 が失敗した場合は、実サーバ real 1.1.1.1 の代わりに real 2.2.2.2 として指定された実サーバがアクティブになります。

コンフィギュレーション例

この付録の各例では、設定に関連する部分のみを示しています。レイヤ 2 およびレイヤ 3 Catalyst スイッチの設定の一部分が含まれる場合もあります。コメント行は # で始まります。**configuration terminal** コマンドを入力して、コンフィギュレーション モードが開始されると、コメント行を設定にペーストすることができます。

vlan コマンドを使用して、スイッチ上の Content Switching Module（CSM; コンテント スイッチング モジュール）の設定で使用されるすべての VLAN（仮想 LAN）が作成されていることを確認します。

MSFC によるクライアント側のルータ モードの設定

ここでは、ルータ モードを設定する設定パラメータの例を示します。

```
module ContentSwitchingModule 5
 vlan 220 server
  ip address 10.20.220.2 255.255.255.0
  alias 10.20.220.1 255.255.255.0

# The servers' default gateway is the alias IP address
# Alias IP addresses are needed any time that you are
# configuring a redundant system.
# However, it is a good practice to always use a
# alias IP address so that a standby CSM can easily
# be added without changes to the IP addressing scheme

!
 vlan 221 client
  ip address 10.20.221.5 255.255.255.0
  gateway 10.20.221.1

# The CSM default gateway in this config is the
# MSFC IP address on that VLAN

!
 serverfarm WEBFARM
  nat server
  no nat client
  real 10.20.220.10
   inservice
  real 10.20.220.20
   inservice
  real 10.20.220.30
   no inservice
!
 vserver WEB
  virtual 10.20.221.100 tcp www
  serverfarm WEBFARM
  persistent rebalance
  inservice

# "persistence rebalance" is effective ONLY when performing
# L7 load balancing (parsing of URLs, cookies, header, ...)
# and only for HTTP 1.1 connections.
# It tells the CSM to parse and eventually make a new
# load balancing decision for each GET within the same
# TCP connection.

interface FastEthernet2/2
 no ip address
 switchport
 switchport access vlan 220

# The above is the port that connects to the real servers

interface FastEthernet2/24
 ip address 10.20.1.1 255.255.255.0

# The above is the interface that connects to the client side network

interface Vlan221
 ip address 10.20.221.1 255.255.255.0

# The above is the MSFC interface for the internal VLAN used
# for MSFC-CSM communication
```

show コマンドの出力

```
Cat6k-2# show module csm 5 arp

Internet Address  Physical Interface  VLAN      Type       Status
------------------------------------------------------------------
 10.20.220.1      00-02-FC-E1-68-EB   220       -ALIAS-    local
 10.20.220.2      00-02-FC-E1-68-EC   220       --SLB--    local
 10.20.220.10     00-D0-B7-A0-81-D8   220       REAL       up(0 misses)
 10.20.221.1      00-02-FC-CB-70-0A   221       GATEWAY    up(0 misses)
 10.20.221.5      00-02-FC-E1-68-EC   221       --SLB--    local
 10.20.220.20     00-D0-B7-A0-81-D8   220       REAL       up(0 misses)
 10.20.220.30     00-D0-B7-A0-81-D8   220       REAL       up(0 misses)
 10.20.221.100    00-02-FC-E1-68-EB   0         VSERVER    local

Cat6k-2# show module csm 5 vlan detail
vlan   IP address       IP mask          type
-------------------------------------------------
220    10.20.220.2      255.255.255.0    SERVER
  ALIASES
  IP address       IP mask
  --------------------------------
  10.20.220.1      255.255.255.0
221    10.20.221.5      255.255.255.0    CLIENT
  GATEWAYS
  10.20.221.1
Cat6k-2#
Cat6k-2# show module csm 5 real

real                  server farm      weight  state          conns/hits
-------------------------------------------------------------------------
10.20.220.10          WEBFARM          8       OPERATIONAL    0
10.20.220.20          WEBFARM          8       OPERATIONAL    0
10.20.220.30          WEBFARM          8       OUTOFSERVICE   0
Cat6k-2#
Cat6k-2# show module csm 5 real detail
10.20.220.10, WEBFARM, state = OPERATIONAL
  conns = 0, maxconns = 4294967295, minconns = 0
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  total conns established = 5, total conn failures = 0
10.20.220.20, WEBFARM, state = OPERATIONAL
  conns = 0, maxconns = 4294967295, minconns = 0
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  total conns established = 5, total conn failures = 0
10.20.220.30, WEBFARM, state = OUTOFSERVICE
  conns = 0, maxconns = 4294967295, minconns = 0
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  total conns established = 0, total conn failures = 0

Cat6k-2#
Cat6k-2# show module csm 5 vserver detail
WEB, type = SLB, state = OPERATIONAL, v_index = 17
  virtual = 10.20.221.100/32:80 bidir, TCP, service = NONE, advertise = FALSE
  idle = 3600, replicate csrp = none, vlan = ALL, pending = 30, layer 4
  max parse len = 2000, persist rebalance = TRUE
  ssl sticky offset = 0, length = 32
  conns = 0, total conns = 10
  Default policy:
    server farm = WEBFARM, backup = <not assigned>
    sticky: timer = 0, subnet = 0.0.0.0, group id = 0
  Policy          Tot matches  Client pkts  Server pkts
  ------------------------------------------------------
  (default)       10           50           50
```

```
Cat6k-2#
Cat6k-2# show module csm 5 stats
Connections Created:        28
Connections Destroyed:      28
Connections Current:        0
Connections Timed-Out:      0
Connections Failed:         0
Server initiated Connections:
      Created: 0, Current: 0, Failed: 0
L4 Load-Balanced Decisions: 27
L4 Rejected Connections:    1
L7 Load-Balanced Decisions: 0
L7 Rejected Connections:
      Total: 0, Parser: 0,
      Reached max parse len: 0, Cookie out of mem: 0,
      Cfg version mismatch: 0, Bad SSL2 format: 0
L4/L7 Rejected Connections:
      No policy: 1, No policy match 0,
      No real: 0, ACL denied 0,
      Server initiated: 0
Checksum Failures:  IP: 0, TCP: 0
Redirect Connections: 0,  Redirect Dropped: 0
FTP Connections:            0
MAC Frames:
      Tx: Unicast: 345, Multicast: 5, Broadcast: 25844,
          Underflow Errors: 0
      Rx: Unicast: 1841, Multicast: 448118, Broadcast: 17,
          Overflow Errors: 0, CRC Errors: 0
```

MSFC によるクライアント側のブリッジ モードの設定

ここでは、ブリッジ モードを設定する設定パラメータの例を示します。

```
module ContentSwitchingModule 5
 vlan 221 client
  ip address 10.20.220.2 255.255.255.0
  gateway 10.20.220.1
!
 vlan 220 server
  ip address 10.20.220.2 255.255.255.0

# Two VLANs with the same IP address are bridged together.

!
 serverfarm WEBFARM
  nat server
  no nat client
  real 10.20.220.10
   inservice
  real 10.20.220.20
   inservice
  real 10.20.220.30
   no inservice
!
 vserver WEB
  virtual 10.20.220.100 tcp www
  serverfarm WEBFARM
  persistent rebalance
  inservice

interface FastEthernet2/2
 no ip address
 switchport
 switchport access vlan 220

# The above is the port that connects to the real servers

interface FastEthernet2/24
 ip address 10.20.1.1 255.255.255.0

# The above is the MSFC interface that connects to the client side network

interface Vlan221
 ip address 10.20.220.1 255.255.255.0

# The above is the MSFC interface for the internal VLAN used
# for MSFC-CSM communication.
# The servers use this IP address as their default gateway
# since the CSM is bridging between the client and server VLANs
```

show コマンドの出力

```
Cat6k-2# show module csm 5 arp

Internet Address  Physical Interface  VLAN      Type       Status
------------------------------------------------------------------
 10.20.220.1      00-02-FC-CB-70-0A   221       GATEWAY    up(0 misses)
 10.20.220.2      00-02-FC-E1-68-EC   221/220   --SLB--    local
 10.20.220.10     00-D0-B7-A0-81-D8   220       REAL       up(0 misses)
 10.20.220.20     00-D0-B7-A0-81-D8   220       REAL       up(0 misses)
 10.20.220.30     00-D0-B7-A0-81-D8   220       REAL       up(0 misses)
 10.20.220.100    00-02-FC-E1-68-EB   0         VSERVER    local
```

プローブの設定

ここでは、プローブを設定する設定パラメータの例を示します。

```
module ContentSwitchingModule 5
 vlan 220 server
  ip address 10.20.220.2 255.255.255.0
  alias 10.20.220.1 255.255.255.0
!
 vlan 221 client
  ip address 10.20.221.5 255.255.255.0
  gateway 10.20.221.1
!
 probe PING icmp
  interval 5
  failed 10
  receive 4

# Interval between the probes is 5 seconds for healthy servers
# while it is 10 seconds for failed servers.
# The servers need to reply within 4 seconds.

!
 probe TCP tcp
  interval 5
  failed 10
  open 4

# The servers need to open the TCP connection within 4 seconds.

!
 probe HTTP http
  request method head url /probe/http_probe.html
  expect status 200 299
  interval 20
  port 80

# The port for the probe is inherited from the vservers.
# The port is necessary in this case, since the same farm
# is serving a vserver on port 80 and one on port 23.
# If the "port 80" parameter is removed, the HTTP probe
# will be sent out on both ports 80 and 23, thus failing
# on port 23 which does not serve HTTP requests.

 probe PING-SERVER-30 icmp
  interval 5
  failed 10
!
 serverfarm WEBFARM
  nat server
  no nat client
  real 10.20.220.10
   inservice
  real 10.20.220.20
   inservice
  real 10.20.220.30
   health probe PING-SERVER-30
   inservice
  probe PING
  probe TCP
  probe HTTP
!
 vserver TELNET
  virtual 10.20.221.100 tcp telnet
  serverfarm WEBFARM
  persistent rebalance
  inservice
!
```

```
 vserver WEB
  virtual 10.20.221.100 tcp www
  serverfarm WEBFARM
  persistent rebalance
  inservice
!
```

show コマンドの出力

```
Cat6k-2# show module csm 5 probe

probe           type    port  interval retries failed  open   receive
------------------------------------------------------------------------
PING            icmp           5        3       10             4
TCP             tcp            5        3       10      4
HTTP            http    80     20       3       300     10     10
PING-SERVER-30  icmp           5        3       10             10

Cat6k-2# show module csm 5 probe detail
probe           type    port  interval retries failed  open   receive
------------------------------------------------------------------------
PING            icmp           5        3       10             4
 real                  vserver         serverfarm      policy          status
 ---------------------------------------------------------------------------
 10.20.220.30:80       WEB             WEBFARM         (default)       OPERABLE
 10.20.220.20:80       WEB             WEBFARM         (default)       OPERABLE
 10.20.220.10:80       WEB             WEBFARM         (default)       OPERABLE
 10.20.220.30:23       TELNET          WEBFARM         (default)       OPERABLE
 10.20.220.20:23       TELNET          WEBFARM         (default)       OPERABLE
 10.20.220.10:23       TELNET          WEBFARM         (default)       OPERABLE
TCP             tcp            5        3       10      4
 real                  vserver         serverfarm      policy          status
 ---------------------------------------------------------------------------
 10.20.220.30:80       WEB             WEBFARM         (default)       OPERABLE
 10.20.220.20:80       WEB             WEBFARM         (default)       OPERABLE
 10.20.220.10:80       WEB             WEBFARM         (default)       OPERABLE
 10.20.220.30:23       TELNET          WEBFARM         (default)       OPERABLE
 10.20.220.20:23       TELNET          WEBFARM         (default)       OPERABLE
 10.20.220.10:23       TELNET          WEBFARM         (default)       OPERABLE
HTTP            http    80     20       3       300     10     10
 Probe Request:  HEAD       /probe/http_probe.html
 Expected Status Codes:
  200 to 299
 real                  vserver         serverfarm      policy          status
 ---------------------------------------------------------------------------
 10.20.220.30:80       WEB             WEBFARM         (default)       OPERABLE
 10.20.220.20:80       WEB             WEBFARM         (default)       FAILED
 10.20.220.10:80       WEB             WEBFARM         (default)       OPERABLE
 10.20.220.30:80       TELNET          WEBFARM         (default)       OPERABLE
 10.20.220.20:80       TELNET          WEBFARM         (default)       FAILED
 10.20.220.10:80       TELNET          WEBFARM         (default)       OPERABLE
PING-SERVER-30  icmp           5        3       10             10
 real                  vserver         serverfarm      policy          status
 ---------------------------------------------------------------------------
 10.20.220.30:80       WEB             WEBFARM         (default)       OPERABLE
 10.20.220.30:23       TELNET          WEBFARM         (default)       OPERABLE

Cat6k-2# show module csm 5 real

real                  server farm     weight  state          conns/hits
------------------------------------------------------------------------
10.20.220.10          WEBFARM         8       OPERATIONAL    0
10.20.220.20          WEBFARM         8       PROBE_FAILED   0
10.20.220.30          WEBFARM         8       OPERATIONAL    0
```

サーバを送信元とする VIP への接続用の送信元 NAT の設定

この例では、サーバが、クライアントのアクセス先と同じ VIP アドレスへのオープン接続を持つ状況を示します。サーバが、サーバどうしでバランスを保つために、送信元 Network Address Translation（NAT; ネットワーク アドレス変換）が必要となります。送信元 NAT を設定するには、仮想サーバ コンフィギュレーションで **vlan** パラメータを使用して、接続が開始される VLAN を識別します。次に、異なるサーバ ファームを使用して、サーバを送信元とする接続を処理します。このサーバ ファーム用に、送信元 NAT が設定されます。送信元 NAT は、クライアントを送信元とする接続には使用されないため、サーバは実クライアントの IP アドレスを記録することができます。

（注） 同じ VLAN 内に位置する送信元および宛先サーバが、サーバ間でロードバランスされた接続をサポートする必要がある場合は、同様の設定を使用する必要があります。

```
module ContentSwitchingModule 5
 vlan 220 server
  ip address 10.20.220.2 255.255.255.0
  alias 10.20.220.1 255.255.255.0
!
 vlan 221 client
  ip address 10.20.221.5 255.255.255.0
  gateway 10.20.221.1
!
 natpool POOL-1 10.20.220.99 10.20.220.99 netmask 255.255.255.0
!
 serverfarm FARM
  nat server
  no nat client
  real 10.20.220.10
   inservice
  real 10.20.220.20
   inservice
  real 10.20.220.30
   inservice
!
 serverfarm FARM2
  nat server
  nat client POOL-1
  real 10.20.220.10
   inservice
  real 10.20.220.20
   inservice
  real 10.20.220.30
   inservice
!
 vserver FROM-CLIENTS
  virtual 10.20.221.100 tcp telnet
  vlan 221
  serverfarm FARM
  persistent rebalance
  inservice
!
 vserver FROM-SERVERS
  virtual 10.20.221.100 tcp telnet
  vlan 220
  serverfarm FARM2
  persistent rebalance
  inservice
```

show コマンドの出力

```
Cat6k-2# show module csm 5 vser
vserver         type  prot virtual                  vlan state        conns
------------------------------------------------------------------------------
FROM-CLIENTS    SLB   TCP  10.20.221.100/32:23      221  OPERATIONAL  1
FROM-SERVERS    SLB   TCP  10.20.221.100/32:23      220  OPERATIONAL  1

Cat6k-2# show module csm 5 conn detail

    prot vlan source               destination          state
------------------------------------------------------------------------
In  TCP  220  10.20.220.10:32858    10.20.221.100:23     ESTAB
Out TCP  220  10.20.220.20:23       10.20.220.99:8193    ESTAB
    vs = FROM-SERVERS, ftp = No, csrp = False

In  TCP  221  10.20.1.100:42443     10.20.221.100:23     ESTAB
Out TCP  220  10.20.220.10:23       10.20.1.100:42443    ESTAB
    vs = FROM-CLIENTS, ftp = No, csrp = False

# The command shows the open connections and how they are translated.
#
# For each connection, both halves of the connection are shown.
# The output for the second half of each connection
# swaps the source and destination IP:port.
#
# The connection originated by server 10.20.220.10 is source-NAT'ed
# and source-PAT'ed (also its L4 source port needs to be translated)
# Its source IP changes from 10.20.220.10 to 10.20.220.99
# Its source L4 port changes from 32858 to 8193

Cat6k-2# show module csm 5 real

real                  server farm      weight  state          conns/hits
-------------------------------------------------------------------------
10.20.220.10          FARM             8       OPERATIONAL    1
10.20.220.20          FARM             8       OPERATIONAL    0
10.20.220.30          FARM             8       OPERATIONAL    0
10.20.220.10          FARM2            8       OPERATIONAL    0
10.20.220.20          FARM2            8       OPERATIONAL    1
10.20.220.30          FARM2            8       OPERATIONAL    0

Cat6k-2# show module csm 5 natpool
nat client POOL-1  10.20.220.99  10.20.220.99  netmask 255.255.255.0

Cat6k-2# show module csm 5 serverfarm

server farm      type     predictor    nat   reals   redirect  bind id
----------------------------------------------------------------------
FARM             SLB      RoundRobin   S     3       0         0
FARM2            SLB      RoundRobin   S,C   3       0         0
```

セッションの持続性（スティッキ性）の設定

ここでは、セッションの持続性またはスティッキ性を設定する設定パラメータの例を示します。

```
module ContentSwitchingModule 5
 vlan 220 server
  ip address 10.20.220.2 255.255.255.0
  alias 10.20.220.1 255.255.255.0
!
 vlan 221 client
  ip address 10.20.221.5 255.255.255.0
  gateway 10.20.221.1
!
 serverfarm WEBFARM
  nat server
  no nat client
  real 10.20.220.10
   inservice
  real 10.20.220.20
   inservice
  real 10.20.220.30
   inservice
!
 sticky 10 netmask 255.255.255.255 timeout 20
!
 sticky 20 cookie yourname timeout 30
!
 vserver TELNET
  virtual 10.20.221.100 tcp telnet
  serverfarm WEBFARM
  persistent rebalance
  inservice
!
 vserver WEB1
  virtual 10.20.221.101 tcp www
  serverfarm WEBFARM
  sticky 20 group 10
  persistent rebalance
  inservice
!
 vserver WEB2
  virtual 10.20.221.102 tcp www
  serverfarm WEBFARM
  sticky 30 group 20
  persistent rebalance
  inservice
!
```

show コマンドの出力

```
Cat6k-2# show module csm 5 sticky group 10

group   sticky-data              real                  timeout
-----------------------------------------------------------------
10      ip 10.20.1.100           10.20.220.10          793

Cat6k-2# show module csm 5 sticky group 20

group   sticky-data              real                  timeout
-----------------------------------------------------------------
20      cookie 4C656B72:861F0395 10.20.220.20          1597


Cat6k-2# show module csm 5 sticky

group   sticky-data              real                  timeout
-----------------------------------------------------------------
20      cookie 4C656B72:861F0395 10.20.220.20          1584
10      ip 10.20.1.100           10.20.220.10          778
```

ルータ モードでのサーバへのダイレクト アクセスの設定

ここでは、ルータ モードを使用して、バックエンド サーバにダイレクト アクセスを行う仮想サーバの設定例を示します。

(注)　ルータ モードでは、仮想サーバがヒットしない接続はいずれも廃棄されます。

```
module ContentSwitchingModule 5
 vlan 220 server
  ip address 10.20.220.2 255.255.255.0
  alias 10.20.220.1 255.255.255.0
!
 vlan 221 client
  ip address 10.20.221.5 255.255.255.0
  gateway 10.20.221.1
  alias 10.20.221.2 255.255.255.0

# The alias IP is only required in redundant configurations
# This is the IP address that the upstream router (the MSFC
# in this case) will use as next-hop to reach the
# backend servers
# See below for the static route added for this purpose.
#
!
 serverfarm ROUTE
  no nat server
  no nat client
  predictor forward

#
# This serverfarm is not load balancing, but is simply
# routing the traffic according to the CSM routing tables
# The CSM routing table in this example is very simple,
# there is just a default gateway and 2 directly attached
# subnets.
#
# The "no nat server" is very important, since you do not
# want to rewrite the destination IP address when
# forwarding the traffic.

!
 serverfarm WEBFARM
  nat server
  no nat client
  real 10.20.220.10
   inservice
  real 10.20.220.20
   inservice
!
 vserver DIRECT-ACCESS
  virtual 10.20.220.0 255.255.255.0 tcp 0
  serverfarm ROUTE
  persistent rebalance
  inservice

# This vserver is listening to all TCP connections destined to the
# serverfarm IP subnet.
# Note: ping to the backend servers will not work with this example

!
```

```
 vserver WEB
  virtual 10.20.221.100 tcp www
  serverfarm WEBFARM
  persistent rebalance
  inservice

interface Vlan221
 ip address 10.20.221.1 255.255.255.0

# vlan221 is the L3 interface on the MSFC that connects to the CSM
# Client requests are being routed by the MSFC, from its other
# interfaces (not shown in this example) to vlan221.

!
ip classless
ip route 10.20.220.0 255.255.255.0 10.20.221.2

# This static route is necessary to allow the MSFC to reach
# the backend servers.
```

show コマンドの出力

```
Cat6k-2# show module csm 5 conn detail

    prot vlan source               destination          state
------------------------------------------------------------------------
In  TCP  221  10.20.1.100:44268     10.20.220.10:23      ESTAB
Out TCP  220  10.20.220.10:23      10.20.1.100:44268    ESTAB
    vs = DIRECT-ACCESS, ftp = No, csrp = False

# The information displayed shows that the CSM is not rewriting any IP addresses while
# forwarding theconnection from VLAN 221 (client) to VLAN 220 (server) This connection
has
# been created because it was destined to the virtual server DIRECT-ACCESS.

Cat6k-2# show module csm 5 vserver detail
WEB, type = SLB, state = OPERATIONAL, v_index = 14
  virtual = 10.20.221.100/32:80 bidir, TCP, service = NONE, advertise = FALSE
  idle = 3600, replicate csrp = none, vlan = ALL, pending = 30, layer 4
  max parse len = 2000, persist rebalance = TRUE
  ssl sticky offset = 0, length = 32
  conns = 0, total conns = 0
  Default policy:
    server farm = WEBFARM, backup = <not assigned>
    sticky: timer = 0, subnet = 0.0.0.0, group id = 0
  Policy          Tot matches  Client pkts  Server pkts
  ----------------------------------------------------
  (default)       0            0            0

DIRECT-ACCESS, type = SLB, state = OPERATIONAL, v_index = 15
  virtual = 10.20.220.0/24:0 bidir, TCP, service = NONE, advertise = FALSE
  idle = 3600, replicate csrp = none, vlan = ALL, pending = 30, layer 4
  max parse len = 2000, persist rebalance = TRUE
  ssl sticky offset = 0, length = 32
  conns = 1, total conns = 1
  Default policy:
    server farm = ROUTE, backup = <not assigned>
    sticky: timer = 0, subnet = 0.0.0.0, group id = 0
  Policy          Tot matches  Client pkts  Server pkts
  ----------------------------------------------------
  (default)       1            48           35
```

サーバ間のロードバランシングされた接続の設定

ここでは、3 つの VLAN（1 つのクライアント VLAN および 2 つのサーバ VLAN）による CSM の設定例を示します。この設定では、サーバ間でロードバランシングされた接続が許可されます。送信元および宛先サーバは、異なる VLAN 内にあるため、送信元 NAT は必要ありません。

```
module ContentSwitchingModule 5
 vlan 220 server
  ip address 10.20.220.2 255.255.255.0
  alias 10.20.220.1 255.255.255.0
!
 vlan 221 client
  ip address 10.20.221.5 255.255.255.0
  gateway 10.20.221.1
!
 vlan 210 server
  ip address 10.20.210.2 255.255.255.0
  alias 10.20.210.1 255.255.255.0
!
 serverfarm TIER-1
  nat server
  no nat client
  real 10.20.210.10
   inservice
  real 10.20.210.20
   inservice
!
 serverfarm TIER-2
  nat server
  no nat client
  real 10.20.220.10
   inservice
  real 10.20.220.20
   inservice
!
 vserver VIP1
  virtual 10.20.221.100 tcp telnet
  vlan 221
  serverfarm TIER-1
  persistent rebalance
  inservice
!
 vserver VIP2
  virtual 10.20.210.100 tcp telnet
  vlan 210
  serverfarm TIER-2
  persistent rebalance
  inservice
!
```

show コマンドの出力

```
Cat6k-2# show module csm 5 arp

Internet Address  Physical Interface  VLAN      Type       Status
------------------------------------------------------------------
 10.20.210.1      00-02-FC-E1-68-EB   210       -ALIAS-    local
 10.20.210.2      00-02-FC-E1-68-EC   210       --SLB--    local
 10.20.210.10     00-D0-B7-A0-68-5D   210       REAL       up(0 misses)
 10.20.210.20     00-D0-B7-A0-68-5D   210       REAL       up(0 misses)
 10.20.220.1      00-02-FC-E1-68-EB   220       -ALIAS-    local
 10.20.220.2      00-02-FC-E1-68-EC   220       --SLB--    local
 10.20.210.100    00-02-FC-E1-68-EB   0         VSERVER    local
 10.20.220.10     00-D0-B7-A0-81-D8   220       REAL       up(0 misses)
 10.20.221.1      00-02-FC-CB-70-0A   221       GATEWAY    up(0 misses)
 10.20.221.5      00-02-FC-E1-68-EC   221       --SLB--    local
 10.20.220.20     00-D0-B7-A0-81-D8   220       REAL       up(0 misses)
 10.20.221.100    00-02-FC-E1-68-EB   0         VSERVER    local

Cat6k-2# show module csm 5 vser

vserver         type  prot virtual                 vlan state        conns
-------------------------------------------------------------------------
VIP1            SLB   TCP  10.20.221.100/32:23     221  OPERATIONAL  1
VIP2            SLB   TCP  10.20.210.100/32:23     210  OPERATIONAL  1

Cat6k-2# show module csm 5 conn detail

    prot vlan source               destination          state
-------------------------------------------------------------------
In  TCP  221  10.20.1.100:44240    10.20.221.100:23     ESTAB
Out TCP  210  10.20.210.10:23      10.20.1.100:44240    ESTAB
    vs = VIP1, ftp = No, csrp = False

In  TCP  210  10.20.210.10:45885   10.20.210.100:23     ESTAB
Out TCP  220  10.20.220.10:23      10.20.210.10:45885   ESTAB
    vs = VIP2, ftp = No, csrp = False

# The previous command shows a connection opened from a client coming in from VLAN 221
# (client is 10.20.1.100). That connection goes to virtual IP address 1 (VIP1) and is
# balanced to 10.20.210.10. Another connection is opened from server 10.20.210.10,
goes to
# VIP2 and is balanced to 10.20.220.10
```

RHI の設定

CSM はいずれの IP サブネットにおいても、仮想サーバをサポートします。仮想サーバが、Multilayer Switch Feature Card（MSFC; マルチレイヤ スイッチ フィーチャ カード）と直接接続していないサブネット内で設定される場合は、この仮想サーバを処理するサーバ ファームの状態に応じて、スタティック ルートを MSFC ルーティング テーブルに注入するよう CSM を設定できます。

また、このメカニズムを障害の回復または Global Server Load Balancing（GSLB; グローバル サーバ ロード バランシング）ソリューションにも使用することができます。この場合、2 つの異なる CSM が同じ VIP のスタティック ルートを注入します。スタティック ルートは、結果的に異なるコストで、特定の位置に再配分できます。

```
module ContentSwitchingModule 5
 vlan 220 server
  ip address 10.20.220.2 255.255.255.0
  alias 10.20.220.1 255.255.255.0
!
 vlan 221 client
  ip address 10.20.221.5 255.255.255.0
  gateway 10.20.221.1
  alias 10.20.221.2 255.255.255.0
```

エイリアス IP は、CSM がアドバタイズされた仮想サーバに到達するために、ネクストホップとして使用するよう MSFC に指示する IP のため、非常に重要です。

```
!
 probe PING icmp
  interval 2
  retries 2
  failed 10
  receive 2
!
 serverfarm WEBFARM
  nat server
  no nat client
  real 10.20.220.10
   inservice
  real 10.20.220.20
   inservice
  probe PING
!
 vserver WEB
  virtual 10.20.250.100 tcp www
  vlan 221

# By default, a virtual server listens to traffic coming in on any VLAN. You can
restrict
# access to a virtual server by defining a specific VLAN. When using Route Health
# Injection, it is required to specify the VLAN for the virtual server. This tells the
CSM
# which next-hop it needs to program in the static route that it will inject in the
MSFC
# routing tables.

serverfarm WEBFARM
  advertise active

# This is the command that tells the CSM to inject the route for this virtual server.
The
# option "active" tells the CSM to remove the route if the backend serverfarm fails.

persistent rebalance
  inservice
```

show コマンドの出力

```
Cat6k-2# show module csm 5 probe detail
probe           type    port  interval retries failed  open   receive
------------------------------------------------------------------------
PING            icmp          2        2       10             2
 real                  vserver         serverfarm      policy          status
 --------------------------------------------------------------------------------
 10.20.220.20:80       WEB             WEBFARM         (default)       OPERABLE
 10.20.220.10:80       WEB             WEBFARM         (default)       OPERABLE

Cat6k-2# show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is 10.20.1.100 to network 0.0.0.0

     10.0.0.0/8 is variably subnetted, 8 subnets, 3 masks
C       10.21.1.0/24 is directly connected, Vlan21
S       10.20.250.100/32 [1/0] via 10.20.221.2, Vlan221

# The static route to 10.20.250.100 has been automatically created by the CSM, since
both
# servers were healthy.

C       10.20.221.0/24 is directly connected, Vlan221
S*   0.0.0.0/0 [1/0] via 10.30.1.100

Cat6k-2# show module csm 5 vser detail
WEB, type = SLB, state = OPERATIONAL, v_index = 14
  virtual = 10.20.250.100/32:80 bidir, TCP, service = NONE, advertise = TRUE
  idle = 3600, replicate csrp = none, vlan = 221, pending = 30, layer 4
  max parse len = 2000, persist rebalance = TRUE
  ssl sticky offset = 0, length = 32
  conns = 0, total conns = 6
  Default policy:
    server farm = WEBFARM, backup = <not assigned>
    sticky: timer = 0, subnet = 0.0.0.0, group id = 0
  Policy          Tot matches  Client pkts  Server pkts
  ----------------------------------------------------
  (default)       6            36           30

# Failing the servers causes the route to be removed This behaviour is configured with
the
# advertise active command.

Cat6k-2# show module csm 5 probe detail
1d20h: %SYS-5-CONFIG_I: Configured from console by vty0 (probe detail
probe           type    port  interval retries failed  open   receive
------------------------------------------------------------------------
PING            icmp          2        2       10             2
 real                  vserver         serverfarm      policy          status
 --------------------------------------------------------------------------------
 10.20.220.20:80       WEB             WEBFARM         (default)       TESTING
 10.20.220.10:80       WEB             WEBFARM         (default)       TESTING

Cat6k-2#
1d20h: %CSM_SLB-6-RSERVERSTATE: Module 5 server state changed: SLB-NETMGT: ICMP health
probe failed for server 10.20.220.20:80 in serverfarm 'WEBFARM'
1d20h: %CSM_SLB-6-RSERVERSTATE: Module 5 server state changed: SLB-NETMGT: ICMP health
probe failed for server 10.20.220.10:80 in serverfarm 'WEBFARM'
```

```
\Cat6k-2#
Cat6k-2# show module csm 5 probe detail
probe           type    port  interval retries failed  open   receive
------------------------------------------------------------------------
PING            icmp          2        2       10             2
 real                   vserver         serverfarm      policy          status
 --------------------------------------------------------------------------------
 10.20.220.20:80       WEB             WEBFARM         (default)       FAILED
 10.20.220.10:80       WEB             WEBFARM         (default)       FAILED
Cat6k-2#

Cat6k-2# show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is 10.20.1.100 to network 0.0.0.0
     10.0.0.0/8 is variably subnetted, 8 subnets, 3 masks
C       10.21.1.0/24 is directly connected, Vlan21
C       10.20.221.0/24 is directly connected, Vlan221
S*   0.0.0.0/0 [1/0] via 10.30.1.100
```

サーバ名の設定

ここでは、サーバ名を使用してサーバとサーバ ファームを関連付ける別の方法を示します。この方法は、複数のサーバ ファームに同一のサーバを関連付ける場合に適しています。ユーザが、1 つのコマンドだけでサーバをすべてのサーバ ファームのローテーションから外すことができるからです。

```
module ContentSwitchingModule 5
 vlan 220 server
  ip address 10.20.220.2 255.255.255.0
  alias 10.20.220.1 255.255.255.0
!
 vlan 221 client
  ip address 10.20.221.5 255.255.255.0
  gateway 10.20.221.1
  alias 10.20.221.2 255.255.255.0
!
 probe PING icmp
  interval 2
  retries 2
  failed 10
  receive 2
!
 probe FTP ftp
  interval 5
  retries 2
  failed 20
  open 3
  receive 3
!
 probe HTTP http
  request method head
  expect status 200 299
  interval 5
  retries 2
  failed 10
  open 2
  receive 2
!
 real SERVER1
  address 10.20.220.10
  inservice
 real SERVER2
  address 10.20.220.20
  inservice
!
 serverfarm FTPFARM
  nat server
  no nat client
  real name SERVER1
   inservice
  real name SERVER2
   inservice
  probe PING
  probe FTP
!
 serverfarm WEBFARM
  nat server
  no nat client
  real name SERVER1
   inservice
  real name SERVER2
   inservice
  probe PING
  probe HTTP
!
```

```
 vserver FTP
  virtual 10.20.221.100 tcp ftp service ftp
  serverfarm FTPFARM
  persistent rebalance
  inservice
!
 vserver WEB
  virtual 10.20.221.100 tcp www
  serverfarm WEBFARM
  persistent rebalance
  inservice
!
```

show コマンドの出力

```
Cat6k-2# show module csm 5 probe detail
probe           type    port  interval retries failed  open   receive
----------------------------------------------------------------------
PING            icmp          2        2       10             2
 real                   vserver         serverfarm      policy          status
 ---------------------------------------------------------------------------
 10.20.220.20:21        FTP             FTPFARM         (default)       OPERABLE
 10.20.220.10:21        FTP             FTPFARM         (default)       OPERABLE
 10.20.220.20:80        WEB             WEBFARM         (default)       OPERABLE
 10.20.220.10:80        WEB             WEBFARM         (default)       OPERABLE
FTP             ftp           5        2       20      3      3
 Expected Status Codes:
  0 to 999
 real                   vserver         serverfarm      policy          status
 ---------------------------------------------------------------------------
 10.20.220.20:21        FTP             FTPFARM         (default)       OPERABLE
 10.20.220.10:21        FTP             FTPFARM         (default)       OPERABLE
HTTP            http          5        2       10      2      2
 Probe Request:  HEAD       /
 Expected Status Codes:
  200 to 299
 real                   vserver         serverfarm      policy          status
 ---------------------------------------------------------------------------
 10.20.220.20:80        WEB             WEBFARM         (default)       OPERABLE
 10.20.220.10:80        WEB             WEBFARM         (default)       OPERABLE

Cat6k-2# show module csm 5 real

real                   server farm      weight  state          conns/hits
-------------------------------------------------------------------------
SERVER1                FTPFARM          8       OPERATIONAL    0
SERVER2                FTPFARM          8       OPERATIONAL    0
SERVER1                WEBFARM          8       OPERATIONAL    0
SERVER2                WEBFARM          8       OPERATIONAL    0

# Taking a server out of service at the server farm level will only take the server
out of
# service for that specific farm

Cat6k-2# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Cat6k-2(config)# module csm 5
Cat6k-2(config-module-csm)# server webfarm
Cat6k-2(config-slb-sfarm)# real name server1
Cat6k-2(config-slb-real)# no inservice
Cat6k-2(config-slb-real)# end
1d20h: %CSM_SLB-6-RSERVERSTATE: Module 5 server state changed: SLB-NETMGT: Configured
server 10.20.220.10:0 to OUT-OF-SERVICE in serverfarm 'WEBFARM'
Cat6k-2#
1d20h: %SYS-5-CONFIG_I: Configured from console by vty0 (10.20.1.100)
Cat6k-2#
Cat6k-2# show module csm 5 real
```

```
real                  server farm      weight  state          conns/hits
--------------------------------------------------------------------------
SERVER1               FTPFARM          8       OPERATIONAL    0
SERVER2               FTPFARM          8       OPERATIONAL    0
SERVER1               WEBFARM          8       OUTOFSERVICE   0
SERVER2               WEBFARM          8       OPERATIONAL    0
Cat6k-2#

# Taking the server out of service at the real server level will take the server out
of
# service for all the server farms

Cat6k-2# confure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Cat6k-2(config)# module csm 5
Cat6k-2(config-module-csm)# real server1
Cat6k(config-slb-module-real)# no inservice
Cat6k(config-slb-module-real)# end
Cat6k-2#
1d20h: %SYS-5-CONFIG_I: Configured from console by vty0 (10.20.1.100)
Cat6k-2# show module csm 5 real

real                  server farm      weight  state          conns/hits
--------------------------------------------------------------------------
SERVER1               FTPFARM          8       OUTOFSERVICE   0
SERVER2               FTPFARM          8       OPERATIONAL    0
SERVER1               WEBFARM          8       OUTOFSERVICE   0
SERVER2               WEBFARM          8       OPERATIONAL    0
Cat6k-2#
```

バックアップ サーバ ファームの設定

ここでは、仮想サーバにバックアップ サーバ ファームを設定する例を示します。プライマリ サーバ ファームのすべてのサーバで障害が発生した場合、CSM がバックアップ サーバ ファームに要求を送り始めます。仮想サーバにスティッキ性が設定されている場合は、sticky オプションによりバックアップ動作を制御できます。

```
module ContentSwitchingModule 5
 vlan 220 server
  ip address 10.20.220.2 255.255.255.0
  alias 10.20.220.1 255.255.255.0
!
 vlan 221 client
  ip address 10.20.221.5 255.255.255.0
  gateway 10.20.221.1
  alias 10.20.221.2 255.255.255.0
!
 vlan 210 server
  ip address 10.20.210.2 255.255.255.0
  alias 10.20.210.1 255.255.255.0
!
 probe PING icmp
  interval 2
  retries 2
  failed 10
  receive 2
!
 real SERVER1
  address 10.20.220.10
  inservice
 real SERVER2
  address 10.20.220.20
  inservice
 real SERVER3
  address 10.20.210.30
  inservice
 real SERVER4
  address 10.20.210.40
  inservice
!
 serverfarm WEBFARM
  nat server
  no nat client
  real name SERVER1
   inservice
  real name SERVER2
   inservice
  probe PING
!
 serverfarm WEBFARM2
  nat server
  no nat client
  real name SERVER3
   inservice
  real name SERVER4
   inservice
  probe PING
!
 vserver WEB
  virtual 10.20.221.100 tcp www
  serverfarm WEBFARM backup WEBFARM2
  persistent rebalance
  inservice
!
```

show コマンドの出力

```
Cat6k-2# show module csm 5 real

real                  server farm      weight  state          conns/hits
--------------------------------------------------------------------------
SERVER1               WEBFARM          8       OPERATIONAL    0
SERVER2               WEBFARM          8       OPERATIONAL    0
SERVER3               WEBFARM2         8       OPERATIONAL    0
SERVER4               WEBFARM2         8       OPERATIONAL    0

# All the servers are shown as operational.

Cat6k-2# show module csm 5 serverfarm detail
WEBFARM, type = SLB, predictor = RoundRobin
  nat = SERVER
  virtuals inservice = 1, reals = 2, bind id = 0, fail action = none
  inband health config: <none>
  retcode map = <none>
  Probes:
    PING, type = icmp
  Real servers:
    SERVER1, weight = 8, OPERATIONAL, conns = 0
    SERVER2, weight = 8, OPERATIONAL, conns = 0
  Total connections = 0

WEBFARM2, type = SLB, predictor = RoundRobin
  nat = SERVER
  virtuals inservice = 1, reals = 2, bind id = 0, fail action = none
  inband health config: <none>
  retcode map = <none>
  Probes:
    PING, type = icmp
  Real servers:
    SERVER3, weight = 8, OPERATIONAL, conns = 0
    SERVER4, weight = 8, OPERATIONAL, conns = 0
  Total connections = 0

Cat6k-2# show module csm 5 vserver detail
WEB, type = SLB, state = OPERATIONAL, v_index = 18
  virtual = 10.20.221.100/32:80 bidir, TCP, service = NONE, advertise = FALSE
  idle = 3600, replicate csrp = none, vlan = ALL, pending = 30, layer 4
  max parse len = 2000, persist rebalance = TRUE
  ssl sticky offset = 0, length = 32
  conns = 0, total conns = 0
  Default policy:
    server farm = WEBFARM, backup = WEBFARM2 (no sticky)
    sticky: timer = 0, subnet = 0.0.0.0, group id = 0
  Policy          Tot matches  Client pkts  Server pkts
  ------------------------------------------------------
  (default)       0            0            0

# No connections have been sent to the virtual server yet.

Cat6k-2# show module csm 5 vserver detail
WEB, type = SLB, state = OPERATIONAL, v_index = 18
  virtual = 10.20.221.100/32:80 bidir, TCP, service = NONE, advertise = FALSE
  idle = 3600, replicate csrp = none, vlan = ALL, pending = 30, layer 4
  max parse len = 2000, persist rebalance = TRUE
  ssl sticky offset = 0, length = 32
  conns = 0, total conns = 14
  Default policy:
    server farm = WEBFARM, backup = WEBFARM2 (no sticky)
    sticky: timer = 0, subnet = 0.0.0.0, group id = 0
  Policy          Tot matches  Client pkts  Server pkts
  ------------------------------------------------------
  (default)       14           84           70
```

```
# A total of 14 connections have been sent to the virtual server and have been
balanced to # the primary server farm. For each connection, the client has sent 6
packets and the # server has sent 5 packets. Two servers are taken out of service

Cat6k-2#
1d21h: %CSM_SLB-6-RSERVERSTATE: Module 5 server state changed: SLB-NETMGT: ICMP health
probe failed for server 10.20.220.10:80 in serverfarm 'WEBFARM'
1d21h: %CSM_SLB-6-RSERVERSTATE: Module 5 server state changed: SLB-NETMGT: ICMP health
probe failed for server 10.20.220.20:80 in serverfarm 'WEBFARM'

Cat6k-2# show module csm 5 serverfarm detail
WEBFARM, type = SLB, predictor = RoundRobin
  nat = SERVER
  virtuals inservice = 1, reals = 2, bind id = 0, fail action = none
  inband health config: <none>
  retcode map = <none>
  Probes:
    PING, type = icmp
  Real servers:
    SERVER1, weight = 8, PROBE_FAILED, conns = 0
    SERVER2, weight = 8, PROBE_FAILED, conns = 0
  Total connections = 0

# The two servers have failed the probe but the CSM has not yet refreshed the ARP
table
# for them, so the servers are not yet shown in the failed state

WEBFARM2, type = SLB, predictor = RoundRobin
  nat = SERVER
  virtuals inservice = 1, reals = 2, bind id = 0, fail action = none
  inband health config: <none>
  retcode map = <none>
  Probes:
    PING, type = icmp
  Real servers:
    SERVER3, weight = 8, OPERATIONAL, conns = 0
    SERVER4, weight = 8, OPERATIONAL, conns = 0
  Total connections = 0

Cat6k-2# show module csm 5 vserver detail
WEB, type = SLB, state = OUTOFSERVICE, v_index = 18
  virtual = 10.20.221.100/32:80 bidir, TCP, service = NONE, advertise = FALSE
  idle = 3600, replicate csrp = none, vlan = ALL, pending = 30, layer 4
  max parse len = 2000, persist rebalance = TRUE
  ssl sticky offset = 0, length = 32
  conns = 0, total conns = 14
  Default policy:
    server farm = WEBFARM, backup = WEBFARM2 (no sticky)
    sticky: timer = 0, subnet = 0.0.0.0, group id = 0
  Policy          Tot matches  Client pkts  Server pkts
  ------------------------------------------------------
  (default)       14           83           70

# The virtual server is displayed as out of service, even if it is configured with a
# backup server farm, which is healthy. This behaviour is useful if the backup server
farm
# is configured as an HTTP redirect server farm to a different site and you are using
some
# DNS-based GSLB method, where some connections are still being directed to the failed
# virtual server.

# If you want the CSM to consider the virtual server healthy and operational if the
backup
# server farm is healthy, you just need to change an environmental variable.

Cat6k-2# show module csm 5 variable
```

```
variable                          value
------------------------------------------------------------------
ARP_INTERVAL                      300
ARP_LEARNED_INTERVAL              14400
ARP_GRATUITOUS_INTERVAL           15
ARP_RATE                          10
ARP_RETRIES                       3
ARP_LEARN_MODE                    1
ARP_REPLY_FOR_NO_INSERVICE_VIP    0
ADVERTISE_RHI_FREQ                10
AGGREGATE_BACKUP_SF_STATE_TO_VS   0
DEST_UNREACHABLE_MASK             0xffff
FT_FLOW_REFRESH_INT               15
GSLB_LICENSE_KEY                  (no valid license)
HTTP_CASE_SENSITIVE_MATCHING      1
MAX_PARSE_LEN_MULTIPLIER          1
NAT_CLIENT_HASH_SOURCE_PORT       0
ROUTE_UNKNOWN_FLOW_PKTS           0
NO_RESET_UNIDIRECTIONAL_FLOWS     0
SYN_COOKIE_INTERVAL               3
SYN_COOKIE_THRESHOLD              5000
TCP_MSS_OPTION                    1460
TCP_WND_SIZE_OPTION               8192
VSERVER_ICMP_ALWAYS_RESPOND       false
XML_CONFIG_AUTH_TYPE              Basic

# The variable that you want to change is AGGREGATE_BACKUP_SF_STATE_TO_VS

Cat6k-2#
1d21h: %CSM_SLB-6-RSERVERSTATE: Module 5 server state changed: SLB-NETMGT: Server
10.20.220.20 failed ARP request
Cat6k-2#

# The CSM has refreshed the ARP entry for 10.20.220.20 which is now reported in the
failed
state.

Cat6k-2# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Cat6k-2(config)# module csm 5
Cat6k-2(config-module-csm)# variable AGGREGATE_BACKUP_SF_STATE_TO_VS 1
Cat6k-2(config-module-csm)# end

1d21h: %SYS-5-CONFIG_I: Configured from console by vty0 (10.20.1.100)

Cat6k-2# show module csm 5 variable

variable                          value
------------------------------------------------------------------
ARP_INTERVAL                      300
ARP_LEARNED_INTERVAL              14400
ARP_GRATUITOUS_INTERVAL           15
ARP_RATE                          10
ARP_RETRIES                       3
ARP_LEARN_MODE                    1
ARP_REPLY_FOR_NO_INSERVICE_VIP    0
ADVERTISE_RHI_FREQ                10
AGGREGATE_BACKUP_SF_STATE_TO_VS   1
DEST_UNREACHABLE_MASK             0xffff
FT_FLOW_REFRESH_INT               15
GSLB_LICENSE_KEY                  (no valid license)
HTTP_CASE_SENSITIVE_MATCHING      1
MAX_PARSE_LEN_MULTIPLIER          1
NAT_CLIENT_HASH_SOURCE_PORT       0
ROUTE_UNKNOWN_FLOW_PKTS           0
NO_RESET_UNIDIRECTIONAL_FLOWS     0
SYN_COOKIE_INTERVAL               3
SYN_COOKIE_THRESHOLD              5000
TCP_MSS_OPTION                    1460
```

```
TCP_WND_SIZE_OPTION             8192
VSERVER_ICMP_ALWAYS_RESPOND     false
XML_CONFIG_AUTH_TYPE            Basic

Cat6k-2# show module csm 5 vserver detail
WEB, type = SLB, state = OPERATIONAL, v_index = 18
  virtual = 10.20.221.100/32:80 bidir, TCP, service = NONE, advertise = FALSE
  idle = 3600, replicate csrp = none, vlan = ALL, pending = 30, layer 4
  max parse len = 2000, persist rebalance = TRUE
  ssl sticky offset = 0, length = 32
  conns = 0, total conns = 14
  Default policy:
    server farm = WEBFARM, backup = WEBFARM2 (no sticky)
    sticky: timer = 0, subnet = 0.0.0.0, group id = 0
  Policy          Tot matches  Client pkts  Server pkts
  ------------------------------------------------------
  (default)       14           83           70

# The virtual server is now shown as operational.

Cat6k-2# show module csm 5 real detail
SERVER1, WEBFARM, state = PROBE_FAILED
  address = 10.20.220.10, location = <NA>
  conns = 0, maxconns = 4294967295, minconns = 0
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  total conns established = 7, total conn failures = 0
SERVER2, WEBFARM, state = FAILED
  address = 10.20.220.20, location = <NA>
  conns = 0, maxconns = 4294967295, minconns = 0
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  total conns established = 7, total conn failures = 0
SERVER3, WEBFARM2, state = OPERATIONAL
  address = 10.20.210.30, location = <NA>
  conns = 0, maxconns = 4294967295, minconns = 0
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  total conns established = 0, total conn failures = 0
SERVER4, WEBFARM2, state = OPERATIONAL
  address = 10.20.210.40, location = <NA>
  conns = 0, maxconns = 4294967295, minconns = 0
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  total conns established = 0, total conn failures = 0
Cat6k-2#

1d21h: %CSM_SLB-6-RSERVERSTATE: Module 5 server state changed: SLB-NETMGT: Server
10.20.220.10 failed ARP request

# The ARP entry for the other server has been refreshed.

Cat6k-2# show module csm 5 real detail
SERVER1, WEBFARM, state = FAILED
  address = 10.20.220.10, location = <NA>
  conns = 0, maxconns = 4294967295, minconns = 0
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  total conns established = 7, total conn failures = 0
SERVER2, WEBFARM, state = FAILED
  address = 10.20.220.20, location = <NA>
  conns = 0, maxconns = 4294967295, minconns = 0
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  total conns established = 7, total conn failures = 0
SERVER3, WEBFARM2, state = OPERATIONAL
  address = 10.20.210.30, location = <NA>
  conns = 0, maxconns = 4294967295, minconns = 0
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  total conns established = 0, total conn failures = 0
SERVER4, WEBFARM2, state = OPERATIONAL
  address = 10.20.210.40, location = <NA>
  conns = 0, maxconns = 4294967295, minconns = 0
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  total conns established = 0, total conn failures = 0
```

```
# So far, each of the servers in the primary server farm have received 7 connections.
New
# connections are now sent only to the backup server farm.

Cat6k-2# show module csm 5 real detail
SERVER1, WEBFARM, state = FAILED
  address = 10.20.220.10, location = <NA>
  conns = 0, maxconns = 4294967295, minconns = 0
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  total conns established = 7, total conn failures = 0
SERVER2, WEBFARM, state = FAILED
  address = 10.20.220.20, location = <NA>
  conns = 0, maxconns = 4294967295, minconns = 0
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  total conns established = 7, total conn failures = 0
SERVER3, WEBFARM2, state = OPERATIONAL
  address = 10.20.210.30, location = <NA>
  conns = 0, maxconns = 4294967295, minconns = 0
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  total conns established = 6, total conn failures = 0
SERVER4, WEBFARM2, state = OPERATIONAL
  address = 10.20.210.40, location = <NA>
  conns = 0, maxconns = 4294967295, minconns = 0
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  total conns established = 6, total conn failures = 0
Cat6k-2#
```

送信元 IP アドレスに基づいたロードバランシングの決定の設定

ここでは、クライアントの送信元 IP アドレスに基づいてロードバランシングを決定する例を示します。この設定では、slb ポリシーを使用する必要があります。

```
module ContentSwitchingModule 5
 vlan 220 server
  ip address 10.20.220.2 255.255.255.0
  alias 10.20.220.1 255.255.255.0
!
 vlan 221 client
  ip address 10.20.221.5 255.255.255.0
  gateway 10.20.221.1
  alias 10.20.221.2 255.255.255.0
!
 probe PING icmp
  interval 2
  retries 2
  failed 10
  receive 2
!
 real SERVER1
  address 10.20.220.10
  inservice
 real SERVER2
  address 10.20.220.20
  inservice
 real SERVER3
  address 10.20.220.30
  inservice
 real SERVER4
  address 10.20.220.40
  inservice
!
 serverfarm WEBFARM
  nat server
  no nat client
  real name SERVER1
   inservice
  real name SERVER2
   inservice
  probe PING
!
 serverfarm WEBFARM2
  nat server
  no nat client
  real name SERVER3
   inservice
  real name SERVER4
   inservice
!
 policy SOURCE-IP-50
  client-group 50
  serverfarm WEBFARM2

# A policy consists of a series of conditions, plus the actions to take if those
# conditions are matched. In this case, the only condition is client-group 50 which
# requires the incoming connection to match the standard access-list 50. The only
action
# to take is to use server farm WEBFARM2 to serve those requests.

!
 vserver WEB
  virtual 10.20.221.100 tcp www
  serverfarm WEBFARM
  persistent rebalance
  slb-policy SOURCE-IP-50
```

```
# Slb-policies associated to a virtual server are always examined in the order in
which
# they are configured. The defintion of the server farm under the virtual server
# configuration is the default policy and is always used as a last resort if no policy
# matches, or if there are no policies configured.

# In this case, incoming requests are processed to see if they match the conditions of
the
# slb-policy SOURCe-IP-50. If they do, then the server farm WEBFARM2 is used,
otherwise
# the default policy is selected (for example, WEBFARM is used).

# If a default server farm is not configured, then connections that do not match any
# policy are dropped.

# This example shows how to configure the IOS standard access list. You can configure
any
# of the 1-99 standard access lists, or you can configure named access lists

inservice
!
access-list 50 permit 10.20.1.100
```

show コマンドの出力

```
Cat6k-2# show module csm 5 vser detail
WEB, type = SLB, state = OPERATIONAL, v_index = 18
  virtual = 10.20.221.100/32:80 bidir, TCP, service = NONE, advertise = FALSE
  idle = 3600, replicate csrp = none, vlan = ALL, pending = 30, layer 4
  max parse len = 2000, persist rebalance = TRUE
  ssl sticky offset = 0, length = 32
  conns = 0, total conns = 0
  Default policy:
    server farm = WEBFARM, backup = <not assigned>
    sticky: timer = 0, subnet = 0.0.0.0, group id = 0
  Policy          Tot matches  Client pkts  Server pkts
  ------------------------------------------------------
  SOURCE-IP-50    0            0            0
  (default)       0            0            0

# This example shows that six connections have matched the slb-policy SOURCE-IP-50.

Cat6k-2# show module csm 5 vser detail
WEB, type = SLB, state = OPERATIONAL, v_index = 18
  virtual = 10.20.221.100/32:80 bidir, TCP, service = NONE, advertise = FALSE
  idle = 3600, replicate csrp = none, vlan = ALL, pending = 30, layer 4
  max parse len = 2000, persist rebalance = TRUE
  ssl sticky offset = 0, length = 32
  conns = 0, total conns = 6
  Default policy:
    server farm = WEBFARM, backup = <not assigned>
    sticky: timer = 0, subnet = 0.0.0.0, group id = 0
  Policy          Tot matches  Client pkts  Server pkts
  ------------------------------------------------------
  SOURCE-IP-50    6            36           30
  (default)       0            0            0

# This example shows that SERVER3 and SERVER4 have received 3 connections each.

Cat6k-2# show module csm 5 real detail
SERVER1, WEBFARM, state = OPERATIONAL
  address = 10.20.220.10, location = <NA>
  conns = 0, maxconns = 4294967295, minconns = 0
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  total conns established = 0, total conn failures = 0
SERVER2, WEBFARM, state = OPERATIONAL
  address = 10.20.220.20, location = <NA>
  conns = 0, maxconns = 4294967295, minconns = 0
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  total conns established = 0, total conn failures = 0
SERVER3, WEBFARM2, state = OPERATIONAL
  address = 10.20.220.30, location = <NA>
  conns = 0, maxconns = 4294967295, minconns = 0
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  total conns established = 3, total conn failures = 0
SERVER4, WEBFARM2, state = OPERATIONAL
  address = 10.20.220.40, location = <NA>
  conns = 0, maxconns = 4294967295, minconns = 0
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  total conns established = 3, total conn failures = 0
Cat6k-2#
```

レイヤ 7 ロードバランシングの設定

ここでは、レイヤ 7 の情報に基づいてロードバランシングを決定する例を示します。この場合、CSM では TCP 接続を終了し、要求をバッファに格納して、要求がポリシー条件に一致するか解析します。ロードバランスが決定されると、CSM は選択されたサーバとの接続を開始して、2 つのフローを 1 つに結合します。

この例の設定では、マップおよびポリシーを使用する必要があります。ポリシーは条件および動作のリストで、すべての条件が真の場合に有効となります。

```
Cat6k-2(config-module-csm)# policy test
Cat6k-2(config-slb-policy)# ?
SLB policy config
  client-group    define policy client group
  cookie-map      define policy cookie map
  default         Set a command to its defaults
  exit            exit slb policy submode
  header-map      define policy header map
  no              Negate a command or set its defaults
  reverse-sticky  define sticky group for reverse traffic
  serverfarm      define policy serverfarm
  set             set policy parameters
  sticky-group    define policy sticky group
  url-map         define policy URL map

# The conditions are:
# -client-group (source IP matches a certain ACL)
# -cookie-map (match based on cookies)
# -header-map (match based on HTTP headers)
# -url-map (match based on URLs)

# The actions are:
# -serverfarm (the most common: use this serverfarm)
# -sticky-group (use sticky)
# -reverse-sticky (use reverse sticky)
# -set (set ip dscp)

\module ContentSwitchingModule 5
 vlan 220 server
  ip address 10.20.220.2 255.255.255.0
  alias 10.20.220.1 255.255.255.0
!
 vlan 221 client
  ip address 10.20.221.5 255.255.255.0
  gateway 10.20.221.1
  alias 10.20.221.2 255.255.255.0
!
 probe PING icmp
  interval 2
  retries 2
  failed 10
  receive 2
!
 map TEST header
  match protocol http header Host header-value www.test.com
!
 map SPORTS url
  match protocol http url /sports/*

# The definition of maps is based on the header and the URL. The URL starts right
after
# the host. For example, in the URL http://www.test.com/sports/basketball/ the URL
portion
# that the URL map applies to is /sports/basketball/.

!
```

```
 real SERVER1
  address 10.20.220.10
  inservice
 real SERVER2
  address 10.20.220.20
  inservice
 real SERVER3
  address 10.20.220.30
  inservice
 real SERVER4
  address 10.20.220.40
  inservice
!
 serverfarm WEBFARM
  nat server
  no nat client
  real name SERVER1
   inservice
  real name SERVER2
   inservice
  probe PING
!
 serverfarm WEBFARM2
  nat server
  no nat client
  real name SERVER3
   inservice
  real name SERVER4
   inservice
!
 policy TEST-SPORTS-50
  url-map SPORTS
  header-map TEST
  client-group 50
  serverfarm WEBFARM2

# Three conditions need to match for this policy to have a match.

!
 vserver WEB
  virtual 10.20.221.100 tcp www
  serverfarm WEBFARM
  persistent rebalance
  slb-policy TEST-SPORTS-50
  inservice
!
# If the three conditions defined in the policy are true then WEBFARM2 is used
otherwise
# WEBFARM is.
```

show コマンドの出力

```
# In this example, 17 requests have matched the policy Of those, 12 requests have not
# matched the policy

Cat6k-2# show module csm 5 vserver detail
WEB, type = SLB, state = OPERATIONAL, v_index = 18
  virtual = 10.20.221.100/32:80 bidir, TCP, service = NONE, advertise = FALSE
  idle = 3600, replicate csrp = none, vlan = ALL, pending = 30, layer 4
  max parse len = 2000, persist rebalance = TRUE
  ssl sticky offset = 0, length = 32
  conns = 0, total conns = 29
  Default policy:
    server farm = WEBFARM, backup = <not assigned>
    sticky: timer = 0, subnet = 0.0.0.0, group id = 0
  Policy          Tot matches  Client pkts  Server pkts
  ------------------------------------------------------
  TEST-SPORTS-50  17           112          95
  (default)       12           82           72

# This example shows that the 29 connections that were load balanced have been load
# balanced at Layer 7. For example, the CSM has to terminate TCP and parse Layer 5
through
# Layer 7 information.

Cat6k-2# show module csm 5 stats
Connections Created:        29
Connections Destroyed:      29
Connections Current:        0
Connections Timed-Out:      0
Connections Failed:         0
Server initiated Connections:
      Created: 0, Current: 0, Failed: 0
L4 Load-Balanced Decisions: 0
L4 Rejected Connections:    0
L7 Load-Balanced Decisions: 29
L7 Rejected Connections:
      Total: 0, Parser: 0,
      Reached max parse len: 0, Cookie out of mem: 0,
      Cfg version mismatch: 0, Bad SSL2 format: 0
L4/L7 Rejected Connections:
      No policy: 0, No policy match 0,
      No real: 0, ACL denied 0,
      Server initiated: 0
Checksum Failures:  IP: 0, TCP: 0
Redirect Connections: 0,  Redirect Dropped: 0
FTP Connections:            0
MAC Frames:
      Tx: Unicast: 359, Multicast: 0, Broadcast: 8,
          Underflow Errors: 0
      Rx: Unicast: 387, Multicast: 221, Broadcast: 1,
          Overflow Errors: 0, CRC Errors: 0
```

HTTP リダイレクトの設定

ここでは、CSM による HTTP リダイレクト メッセージの送信の設定例を示します。

```
# This configuration represents the configuration of site A

module ContentSwitchingModule 6
 vlan 211 client
  ip address 10.20.211.2 255.255.255.0
  gateway 10.20.211.1
!
 vlan 210 server
  ip address 10.20.210.1 255.255.255.0
!
 map SPORTMAP url
  match protocol http url /sports*
!
 serverfarm REDIRECTFARM
  nat server
  no nat client
  redirect-vserver WWW2
   webhost relocation www2.test.com 301
   inservice
!
 serverfarm WWW1FARM
  nat server
  no nat client
  real 10.20.210.10
   inservice
  real 10.20.210.20
   inservice
!
 policy SPORTPOLICY
  url-map SPORTMAP
  serverfarm REDIRECTFARM
!
 vserver WWW1VIP
  virtual 10.20.211.100 tcp www
  serverfarm WWW1FARM
  persistent rebalance
  slb-policy SPORTPOLICY
  inservice

# This configuration represents the configuration of site B

module ContentSwitchingModule 7
 vlan 221 client
  ip address 10.20.221.2 255.255.255.0
  gateway 10.20.221.1
!
 vlan 220 server
  ip address 10.20.220.1 255.255.255.0
!
 serverfarm WWW2FARM
  nat server
  no nat client
  real 10.20.220.10
   inservice
  real 10.20.220.20
   inservice
!
 vserver WWW2VIP
  virtual 10.20.221.100 tcp www
  serverfarm WWW2FARM
  persistent rebalance
  inservice
```

show コマンドの出力

```
# To test the configuration, the first nine requests are sent to www1.test.com
requesting
# the home page "/." The 10th request is sent to http://www1.test.com/sports/.

Cat6k-2# show module csm 6 vser deta
WWW1VIP, type = SLB, state = OPERATIONAL, v_index = 11
  virtual = 10.20.211.100/32:80 bidir, TCP, service = NONE, advertise = FALSE
  idle = 3600, replicate csrp = none, vlan = ALL, pending = 30
  max parse len = 2000, persist rebalance = TRUE
  ssl sticky offset = 0, length = 32
  conns = 0, total conns = 10
  Default policy:
    server farm = WWW1FARM, backup = <not assigned>
    sticky: timer = 0, subnet = 0.0.0.0, group id = 0
  Policy          Tot Conn     Client pkts  Server pkts
  ------------------------------------------------------
  SPORTPOLICY     1            3            1
  (default)       9            45           45

Cat6k-2# show module csm 7 vser detail
WWW2VIP, type = SLB, state = OPERATIONAL, v_index = 26
  virtual = 10.20.221.100/32:80 bidir, TCP, service = NONE, advertise = FALSE
  idle = 3600, replicate csrp = none, vlan = ALL, pending = 30
  max parse len = 2000, persist rebalance = TRUE
  ssl sticky offset = 0, length = 32
  conns = 0, total conns = 1
  Default policy:
    server farm = WWW2FARM, backup = <not assigned>
    sticky: timer = 0, subnet = 0.0.0.0, group id = 0
  Policy          Tot Conn     Client pkts  Server pkts
  ------------------------------------------------------
  (default)       1            5            5

# Nine requests have matched the default policy for www1.test.com so they have been
served
# by WWW1FARM. One request has matched the policy SPORTPOLICY and has been redirected
to
# the second site that has then served the request.

# The following is an example of the request that was sent to www1.cisco.com asking
for
# /sports/.


10.20.1.100.34589 > 10.20.211.100.80: P 1:287(286) ack 1 win 5840 (DF)
0x0000   4500 0146 763c 4000 4006 da85 0a14 0164        E..Fv<@.@......d
0x0010   0a14 d364 871d 0050 ec1d 69e6 7b57 aead        ...d...P..i.{W..
0x0020   5018 16d0 96b2 0000 4745 5420 2f73 706f        P.......GET./spo
0x0030   7274 732f 2048 5454 502f 312e 310d 0a43        rts/.HTTP/1.1..C
0x0040   6f6e 6e65 6374 696f 6e3a 204b 6565 702d        onnection:.Keep-
0x0050   416c 6976 650d 0a55 7365 722d 4167 656e        Alive..User-Agen
0x0060   743a 204d 6f7a 696c 6c61 2f35 2e30 2028        t:.Mozilla/5.0.(
0x0070   636f 6d70 6174 6962 6c65 3b20 4b6f 6e71        compatible;.Konq
0x0080   7565 726f 722f 322e 322d 3131 3b20 4c69        ueror/2.2-11;.Li
0x0090   6e75 7829 0d0a 4163 6365 7074 3a20 7465        nux)..Accept:.te
0x00a0   7874 2f2a 2c20 696d 6167 652f 6a70 6567        xt/*,.image/jpeg
0x00b0   2c20 696d 6167 652f 706e 672c 2069 6d61        ,.image/png,.ima
0x00c0   6765 2f2a 2c20 2a2f 2a0d 0a41 6363 6570        ge/*,.*/*..Accep
0x00d0   742d 456e 636f 6469 6e67 3a20 782d 677a        t-Encoding:.x-gz
0x00e0   6970 2c20 677a 6970 2c20 6964 656e 7469        ip,.gzip,.identi
0x00f0   7479 0d0a 4163 6365 7074 2d43 6861 7273        ty..Accept-Chars
0x0100   6574 3a20 416e 792c 2075 7466 2d38 2c20        et:.Any,.utf-8,.
0x0110   2a0d 0a41 6363 6570 742d 4c61 6e67 7561        *..Accept-Langua
0x0120   6765 3a20 656e 5f55 532c 2065 6e0d 0a48        ge:.en_US,.en..H
0x0130   6f73 743a 2077 7777 312e 7465 7374 2e63        ost:.www1.test.c
0x0140   6f6d 0d0a 0d0a                                 om....
```

```
# The following example is the message that the client has received back from
# www1.cisco.com. This message is the HTTP redirect message generated by the CSM

10.20.211.100.80 > 10.20.1.100.34589: FP 1:56(55) ack 287 win 2048 (DF)
0x0000   4500 005f 763c 4000 3e06 dd6c 0a14 d364        E.._v<@.>..l...d
0x0010   0a14 0164 0050 871d 7b57 aead ec1d 6b04        ...d.P..{W....k.
0x0020   5019 0800 8b1a 0000 4854 5450 2f31 2e30        P.......HTTP/1.0
0x0030   2033 3031 2046 6f75 6e64 200d 0a4c 6f63        .301.Found...Loc
0x0040   6174 696f 6e3a 2068 7474 703a 2f2f 7777        ation:.http://ww
0x0050   7732 2e74 6573 742e 636f 6d0d 0a0d 0a          w2.test.com....

# The redirect location sent back to the client matches exactly the string configured
with
# the webhost relocation www2.test.com 301 command because the client was browsing
# www1.test.com/sports/ and is redirected to www2.test.com/.

# In some cases this might not be the desired behaviour and there might be the need to
# preserve the original URL that the browser requested.

# To preseerve the URL that the browser requested, you can use the %p parameter as
part of
# the redirect string.

# The configuration would then appear as:

#   serverfarm REDIRECTFARM
#     nat server
#     no nat client
#     redirect-vserver WWW2
#       webhost relocation www2.test.com/%p
#       inservice

# The following example shows the resulting redirect message which is sent back to the
# client:

10.20.211.100.80 > 10.20.1.100.34893: FP 1:64(63) ack 329 win 2048 (DF)
0x0000   4500 0067 7d95 4000 3e06 d60b 0a14 d364        E..g}.@.>......d
0x0010   0a14 0164 0050 884d 7093 b53b 4e0b e8a8        ...d.P.Mp..;N...
0x0020   5019 0800 2800 0000 4854 5450 2f31 2e30        P...(...HTTP/1.0
0x0030   2033 3032 2046 6f75 6e64 200d 0a4c 6f63        .302.Found...Loc
0x0040   6174 696f 6e3a 2068 7474 703a 2f2f 7777        ation:.http://ww
0x0050   7732 2e74 6573 742e 636f 6d2f 7370 6f72        w2.test.com/spor
0x0060   7473 2f0d 0a0d 0a                              ts/....

# In other cases, you may need to redirect an HTTP request to an HTTPS VIP, on the
same or
# on a remote CSM. In that case, the URL request must change from http:// to https://
# You can do this by using the parameter ssl 443

# The configuration would then be as follows:

#   serverfarm REDIRECTFARM
#     nat server
#     no nat client
#     redirect-vserver WWW2
#       webhost relocation www2.test.com/%p
#       ssl 443
#       inservice

# The following is the resulting redirect message sent back to the client.

10.20.211.100.80 > 10.20.1.100.34888: FP 1:65(64) ack 329 win 2048 (DF)
0x0000   4500 0068 2cda 4000 3e06 26c6 0a14 d364        E..h,.@.>.&....d
0x0010   0a14 0164 0050 8848 7088 b087 21e5 a627        ...d.P.Hp...!..'
0x0020   5019 0800 f39e 0000 4854 5450 2f31 2e30        P.......HTTP/1.0
0x0030   2033 3032 2046 6f75 6e64 200d 0a4c 6f63        .302.Found...Loc
0x0040   6174 696f 6e3a 2068 7474 7073 3a2f 2f77        ation:.https://w
0x0050   7777 322e 7465 7374 2e63 6f6d 2f73 706f        ww2.test.com/spo
0x0060   7274 732f 0d0a 0d0a                            rts/....
```

APPENDIX B

トラブルシューティングとシステム メッセージ

この付録では、Content Switching Module（CSM; コンテント スイッチング モジュール）のトラブルシューティングとシステム メッセージについて説明します。

トラブルシューティング

CSM が使用不能の場合、モジュールは Address Resolution Protocol（ARP）要求には応答していても、ping には応答しません。

システム メッセージ

ここでは、CSM でサポートされるシステム ログ（Syslog）メッセージを示します。

Cisco IOS ソフトウェアのメッセージ ログには、次の構文による警告レベルが含まれます。

`CSM_SLB_level-code`

表 B-1 に、レベル コードを示します。

表 B-1　エラー メッセージ レベル コード

メッセージ レベル	コード
LOG_EMERG	0 /* システムが使用不可能 */
LOG_ALERT	1 /* ただちに対処が必要 */
LOG_CRIT	2 /* 危険な状態 */
LOG_ERR	3 /* エラー状態 */
LOG_WARNING	4 /* 警告状態 */
LOG_NOTICE	5 /* 正常だが注意を要する状態 */
LOG_INFO	6 /* 情報メッセージ */
LOG_DEBUG	7 /* デバッグレベルのメッセージ */

エラー メッセージ `CSM_SLB-3-IDB_ERROR Unknown error occurred while configuring IDB`

説明　Multilayer Switch Feature Card（MSFC; マルチレイヤ スイッチ フィーチャ カード）は、CSM の内部インターフェイスを作成できませんでした。

対処方法　このバージョンの MSFC または IDPROM が、正常にプログラムされていません。MSFC または IDPROM を再プログラムしてください。

エラー メッセージ `CSM_SLB-3-OUTOFMEM Module [dec] memory error`

説明　これは、コントロール モジュールのメモリの一般的な問題です。メモリの問題は継続すると、さらに深刻な CSM の動作上の問題に発展する可能性があります。

対処方法　メモリ チェックを実行するか、またはメモリ容量を増やしてください。

エラー メッセージ `CSM_SLB-3-PORTCHANNEL Portchannel allocation failed for module [dec]`

説明　この問題は、設定よりも多くの CSM モジュールがシャーシに挿入されている場合、またはモジュールを挿入したスロット番号が想定より上の場合に起こります。

対処方法　CSM モジュールを下の番号のスロットに移動して、問題を解決してください。

エラー メッセージ `CSM_SLB-3-RELOAD Module [dec] configuration reload failed`

説明　MSFC は、オンライン化された CSM モジュールに既存の設定をリロードできませんでした。この問題の原因は、CLI（コマンドライン インターフェイス）による CSM のエラー チェックである可能性があります。

対処方法　診断上の障害またはバージョンの不一致など CSM モジュールのステータスを確認してください。

エラー メッセージ CSM_SLB-3-UNEXPECTED Module [dec] unexpected error
CSM_SLB-3-REDUNDANCY Module [dec] FT error
CSM_SLB-4-REDUNDANCY_WARN Module [dec] FT warning
CSM_SLB-6-REDUNDANCY_INFO Module %d FT info
CSM_SLB-3-ERROR Module [dec] error
CSM_SLB-4-WARNING Module [dec] warning
CSM_SLB-6-INFO Module [dec] info

説明　これらのメッセージは、エラーまたは警告メッセージに関する一般的な見出しです。追加の詳細については、情報ストリングに示されます。

対処方法　なし。

エラー メッセージ CSM_SLB-3-VERMISMATCH Module [dec] image version mismatch

説明　これは、MSFC と CSM コード間のバージョンの不一致です。この状態は、MSFC のソフトウェア リリースが Release 12.1(8)EX より前の場合、または CSM のソフトウェア リリースが Release 2.1(1) より前の場合にのみ発生します。

対処方法　CSM のオンライン化が可能な CSM リリースに一致するように、MSFC のリリースをアップグレードまたはダウングレードしてください。

エラー メッセージ CSM_SLB-4-ARPCONFIG Module [dec] ARP configuration error

説明　スタティックな ARP 設定の作成または削除におけるエラーです。

対処方法　ARP 設定を再確認してください。

エラー メッセージ CSM_SLB-4-ERRPARSING Module [dec] configuration warning
SLB-REGEX: Syntactiv error in regular expression <x>.
SLB-REGEX: Parse error in regular expression <x>.

説明　これらは、URL、Cookie、またはヘッダーの正規表現照合に関する構文エラーチェック メッセージです。

対処方法　入力照合ストリングを確認してください。

エラー メッセージ CSM_SLB-4-INVALIDID Module [dec] invalid ID
CSM_SLB-4-DUPLICATEID Module [dec] duplicate ID

説明　これらは、2 つのモジュール間で一方のモジュールがもう一方を呼び出している場合のエラーチェック メッセージです。

対処方法　これらのエラーが発生しないようにする CLI レベルでエラーを確認してください。

エラー メッセージ CSM_SLB-4-PROBECONFIG Module [dec] probe configuration error

説明　CSM には、指定されたプローブ設定をサポートできる十分なメモリがありません。

対処方法　サーバ ファームからプローブをいくつか削除してください。

エラー メッセージ CSM_SLB-4-REGEXMEM Module [dec] regular expression memory error
SLB-LCSC: Error detected while downloading URL configuration for vserver %s.
SLB-LCSC: Error detected while downloading COOKIE policy map for vserver <x>.
SLB-LCSC: Error detected while downloading COOKIE <x> for vserver <x>.
SLB-LCSC: There was an error downloading the configuration to hardware
SLB-LCSC: due to insufficient memory. Use the 'show ips slb memory'
SLB-LCSC: command to gather information about memory usage.

説明　これらのエラーは、複合的な URL、Cookie、またはヘッダー照合表現を設定した場合に発生する可能性があります。CSM では、ストリング照合を計算するスペースが限定されています。現在のところ、仮想サーバごとに 10 個のキーワード（たとえば、「name*」など）までが許容されています。

対処方法　この問題に対処するには、文字列を結合（または削除）してください。

エラー メッセージ CSM_SLB-4-TOPOLOGY Module [dec] warning

説明　CSM は、ネットワーク内で「ブリッジ ループ」を検出しています。

対処方法　ネットワーク内にある複数の CSM のブリッジング装置およびブリッジ モードの設定を確認してください。

エラー メッセージ CSM_SLB-4-VERWILDCARD Received CSM-SLB module version wildcard on slot

説明　前出のエラー メッセージで説明されたイメージ バージョンの不一致状態に対処するために、デバッグ コマンドを CSM コンソールに入力すると、CSM はこのメッセージを送信します。

対処方法　このエラーは、デバッグ状態であるということだけです。

エラー メッセージ SLB-DIAG: WatchDog task not responding.
SLB-DIAG: Fatal Diagnostic Error %x, Info %x.
SLB-DIAG: Diagnostic Warning %x, Info %x.

説明　ボードのブート手順中に、さまざまな診断上の問題が発生しました。

対処方法　CSM のハードウェア故障であるか、またはフラッシュ メモリ内のソフトウェア破損であるかを確認してください。

エラー メッセージ SLB-FT: Heartbeat intervals are not identical between ft pair.
SLB-FT: heartbeat interval is identical again
SLB-FT: The configurations are not identical between the members of the fault tolerant pair.

説明　これらのエラーは、2 つの冗長 CSM モジュール間の不適切な構成により発生します。

対処方法　フォールトトレラント コンフィギュレーションの属性および実サーバとサーバファームのコンフィギュレーションを確認してください。

エラー メッセージ SLB-FT: Standby is not monitoring active now.

説明　この問題は、CSM の 2 つのバージョン間のフォールト トレランス プロトコルのバージョンの不一致により発生します。プライマリ CSM が故障している場合も、スタンバイ CSM はスタンバイのままで、アクティブ CSM として引き継ぐことはありません。このような状況では、CSM はヒットレス（HA）アップグレードをサポートしません。

対処方法　フォールト トレランス プロトコルのバージョンが一致していることを確認してください。

CSM XML の DTD

この Document Type Definition（DTD）を使用して、Content Switching Module（CSM; コンテント スイッチング モジュール）を「XML インターフェイスの設定」（p.8-27）に記載されているように設定することができます。

CSM XML の DTD は、次のとおりです。

```
<!--
/*
 *  cisco_csm.dtd - XML DTD for CSM 4.2
 *
 *  January 2005  Paul Mathison
 *
 *  Copyright (c) 2002, 2003-2005 by cisco Systems, Inc.
 *  All rights reserved
 */
-->

<!--
  Notes:
  Each element refers to a particular IOS CLI command.
  Each attribute refers to a command parameter.
  Except where noted, all "name" attributes are strings of length
    1 to 15, with no whitespace.
  IP address and mask attributes use standard "x.x.x.x" format.
-->

<!--
************************************************************
  Elements and attributes required by various other elements
************************************************************
-->

<!ELEMENT inservice EMPTY>
<!ATTLIST inservice
  sense (yes | no) #IMPLIED
>

<!ELEMENT inservice_standby EMPTY>
<!ATTLIST inservice_standby
  sense (yes | no) #IMPLIED
>

<!--
  backup_name is a string of length 1 to 15
  backup_sticky default is "no"
-->
<!ELEMENT serverfarm_ref EMPTY>
```

```
<!ATTLIST serverfarm_ref
  sense         (yes | no) #IMPLIED
  name          CDATA      #REQUIRED
  backup_name   CDATA      #IMPLIED
  backup_sticky (yes | no) #IMPLIED
>

<!--
  value is between 1 and 4294967295
-->
<!ELEMENT maxconns EMPTY>
<!ATTLIST maxconns
  sense (yes | no) #IMPLIED
  value NMTOKEN    #REQUIRED
>

<!--
  id is between 1 and 255
-->
<!ELEMENT reverse_sticky EMPTY>
<!ATTLIST reverse_sticky
  sense (yes | no) #IMPLIED
  id    NMTOKEN    #REQUIRED
>

<!--
*************************************************************
  Elements and attributes required for env_variable
*************************************************************
-->

<!--
  name is a string of length 1 to 31
  expression is a string of length 0 to 127
-->
<!ELEMENT env_variable EMPTY>
<!ATTLIST env_variable
  sense      (yes | no) #IMPLIED
  name       CDATA      #REQUIRED
  expression CDATA      #REQUIRED
>

<!--
*************************************************************
  Elements and attributes required for owner
*************************************************************
-->
<!--
*************************************************************
  Elements and attributes required for owner_conn.
  This is READ ONLY
*************************************************************
-->

<!ELEMENT owner_conn EMPTY>
<!ATTLIST owner_conn
  connection     NMTOKEN    #IMPLIED
  total_conns NMTOKEN    #IMPLIED
  max_conns_dropped    NMTOKEN    #IMPLIED
>


<!--
  string is of length 1 to 200
-->
<!ELEMENT billing_info EMPTY>
<!ATTLIST billing_info
  sense  (yes | no) #IMPLIED
  string CDATA      #REQUIRED
>
```

```
<!--
  string is of length 1 to 200
-->
<!ELEMENT contact_info EMPTY>
<!ATTLIST contact_info
  sense  (yes | no) #IMPLIED
  string CDATA      #REQUIRED
>

<!ELEMENT owner (maxconns?, billing_info?, contact_info?, owner_conn?)>
<!ATTLIST owner
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>

<!--
*****************************************************************
  Elements and attributes required for xml_vlan. This is READ ONLY.
*****************************************************************
-->
<!ELEMENT xml_vlan EMPTY>
<!ATTLIST xml_vlan
  id        NMTOKEN    #IMPLIED
  ipaddress NMTOKEN    #IMPLIED
>

<!--
*********************************************************************
  Elements and attributes required for xml_client. This is READ ONLY.
*********************************************************************
-->
<!ELEMENT xml_client EMPTY>
<!ATTLIST xml_client
  list  NMTOKEN #IMPLIED
>

<!--
*********************************************************************
  Elements and attributes required for xml_connections.
  This is READ ONLY.
*********************************************************************
-->
<!ELEMENT xml_connections EMPTY>
<!ATTLIST xml_connections
  current NMTOKEN #REQUIRED
  total NMTOKEN #REQUIRED
  failed NMTOKEN #REQUIRED
  security_failed NMTOKEN #REQUIRED
  total_requests NMTOKEN #REQUIRED
  failed_requests NMTOKEN #REQUIRED
>

<!--
*********************************************************************
  Elements and attributes required for xml_stats.
  This is READ ONLY.
*********************************************************************
-->

<!ELEMENT xml_stats (xml_vlan?, xml_client?, xml_connections?)>
<!ATTLIST xml_stats
  status CDATA #REQUIRED
  port  NMTOKEN #REQUIRED
>
```

```
<!--
************************************************************
  Elements and attributes required for vlan
************************************************************
-->

<!ELEMENT vlan_address EMPTY>
<!ATTLIST vlan_address
  sense     (yes | no) #IMPLIED
  ipaddress NMTOKEN    #REQUIRED
  ipmask    NMTOKEN    #REQUIRED
>

<!ELEMENT gateway EMPTY>
<!ATTLIST gateway
  sense     (yes | no) #IMPLIED
  ipaddress NMTOKEN    #REQUIRED
>

<!--
  gateway uses standard x.x.x.x format
-->
<!ELEMENT route EMPTY>
<!ATTLIST route
  sense     (yes | no) #IMPLIED
  ipaddress NMTOKEN    #REQUIRED
  ipmask    NMTOKEN    #REQUIRED
  gateway   NMTOKEN    #REQUIRED
>

<!ELEMENT alias EMPTY>
<!ATTLIST alias
  sense     (yes | no) #IMPLIED
  ipaddress NMTOKEN    #REQUIRED
  ipmask    NMTOKEN    #REQUIRED
>

<!--
  id is between 2 and 4094
  Maximum of 7 gateways per vlan
  Maximum of 4095 routes per vlan
  Maximum of 255 aliases per vlan
  Global maximum of 255 unique vlan_addresses
  Global maximum of 255 vlan gateways (including routed gateways)
-->
<!ELEMENT vlan (vlan_address?, gateway*, route*, alias*)>
<!ATTLIST vlan
  sense (yes | no)        #IMPLIED
  id    NMTOKEN           #REQUIRED
  type  (client | server) #REQUIRED
>

<!--
************************************************************
  Elements and attributes required for script_file, script_task
  and script
************************************************************
-->

<!--
  url is a string of length 1 to 200
-->
<!ELEMENT script_file EMPTY>
<!ATTLIST script_file
  sense        (yes | no)    #IMPLIED
  url          CDATA         #REQUIRED
>
```

```
<!--
  id is between 1 and 100
  name is a string of length 1 to 31
  arguments is a string of length 0 to 199
  num_runns, last_exit_code, status_string, start_time,
  end_time,  last_error_line, last_error,
  last_error_info are READ ONLY fields

-->
<!ELEMENT script_task EMPTY>
<!ATTLIST script_task
  sense            (yes | no) #IMPLIED
  id               NMTOKEN    #REQUIRED
  name             CDATA      #REQUIRED
  arguments        CDATA      #IMPLIED
  num_runs         NMTOKEN    #IMPLIED
  last_exit_code   NMTOKEN    #IMPLIED
  status_string    CDATA      #IMPLIED
  start_time       CDATA      #IMPLIED
  end_time         CDATA      #IMPLIED
  last_error_line  NMTOKEN    #IMPLIED
  last_error       CDATA      #IMPLIED
  last_error_info  CDATA      #IMPLIED
>

<!--
  This is READ ONLY.
-->
<!ELEMENT script EMPTY>
<!ATTLIST script
  name         CDATA         #IMPLIED
  url          CDATA         #IMPLIED
  size         NMTOKEN       #IMPLIED
  load_time    CDATA         #IMPLIED
  code         CDATA         #IMPLIED
>

<!--
*************************************************************
  Elements and attributes required for probe
*************************************************************
-->

<!--
  value is between 2 and 65535 (default is 300)
-->
<!ELEMENT probe_failed EMPTY>
<!ATTLIST probe_failed
  sense (yes | no) #IMPLIED
  value NMTOKEN    #REQUIRED
>

<!--
  value is between 2 and 65535 (default is 120)
-->
<!ELEMENT probe_interval EMPTY>
<!ATTLIST probe_interval
  sense (yes | no) #IMPLIED
  value NMTOKEN    #REQUIRED
>

<!--
  value is between 0 and 65535 (default is 3)
-->
<!ELEMENT probe_retries EMPTY>
<!ATTLIST probe_retries
  sense (yes | no) #IMPLIED
  value NMTOKEN    #REQUIRED
>
```

```
<!--
  value is between 1 and 65535 (default 10)
-->
<!ELEMENT probe_open EMPTY>
<!ATTLIST probe_open
  sense (yes | no) #IMPLIED
  value NMTOKEN    #REQUIRED
>

<!--
  value is between 1 and 65535 (default 10)
-->
<!ELEMENT probe_receive EMPTY>
<!ATTLIST probe_receive
  sense (yes | no) #IMPLIED
  value NMTOKEN    #REQUIRED
>

<!--
  value is between 1 and 65535
-->
<!ELEMENT probe_port EMPTY>
<!ATTLIST probe_port
  sense (yes | no) #IMPLIED
  value NMTOKEN    #REQUIRED
>

<!--
  string is of length 1 to 64
-->
<!ELEMENT probe_domain EMPTY>
<!ATTLIST probe_domain
  sense  (yes | no) #IMPLIED
  string CDATA      #REQUIRED
>

<!ELEMENT probe_address EMPTY>
<!ATTLIST probe_address
  sense     (yes | no) #IMPLIED
  ipaddress NMTOKEN    #REQUIRED
  mode      (transparent | routed) "transparent"
>

<!ELEMENT probe_expect_address EMPTY>
<!ATTLIST probe_expect_address
  sense     (yes | no) #IMPLIED
  ipaddress NMTOKEN    #REQUIRED
>

<!--
  expression is a string of length 1 to 200
-->
<!ELEMENT probe_header EMPTY>
<!ATTLIST probe_header
  sense      (yes | no) #IMPLIED
  name       CDATA      #REQUIRED
  expression CDATA      #REQUIRED
>

<!--
  user is a string of length 1 to 15
  password is a string of length 1 to 15
-->
<!ELEMENT probe_credentials EMPTY>
<!ATTLIST probe_credentials
  sense    (yes | no) #IMPLIED
  user     CDATA      #REQUIRED
  password CDATA      ""
>
```

```
<!--
  url is a string of length 1 to 200
-->
<!ELEMENT probe_request EMPTY>
<!ATTLIST probe_request
  sense  (yes | no)   #IMPLIED
  method (get | head) #REQUIRED
  url    CDATA        "/"
>

<!--
  min_code is between 0 and 999
  max_code default is match min_code
-->
<!ELEMENT probe_expect_status EMPTY>
<!ATTLIST probe_expect_status
  sense    (yes | no) #IMPLIED
  min_code NMTOKEN    #REQUIRED
  max_code NMTOKEN    #IMPLIED
>

<!--
  name is a string of length 1 to 31
  arguments is a string of length 0 to 199
-->
<!ELEMENT script_ref EMPTY>
<!ATTLIST script_ref
  sense     (yes | no) #IMPLIED
  name      CDATA      #REQUIRED
  arguments CDATA      #IMPLIED
>

<!--
  secret is a string of length 1 to 32
-->
<!ELEMENT probe_secret EMPTY>
<!ATTLIST probe_secret
  sense  (yes | no) #IMPLIED
  secret CDATA      #REQUIRED
>

<!--
  Maximum of 255 probe_headers per http_probe
  probe_address must use mode "routed"
-->
<!ELEMENT http_probe (probe_failed?, probe_interval?, probe_retries?,
                      probe_open?, probe_receive?, probe_port?, probe_address?,
                      probe_request?, probe_credentials?, probe_header*,
                      probe_expect_status*)
>

<!--
  Maximum of 255 probe_expect_addresses per dns_probe
  probe_address must use mode "routed"
-->
<!ELEMENT dns_probe (probe_failed?, probe_interval?, probe_retries?,
                     probe_receive?, probe_port?, probe_address?, probe_domain?,
                     probe_expect_address*)
>

<!--
  probe_address must use mode "transparent"
-->
<!ELEMENT icmp_probe (probe_failed?, probe_interval?, probe_retries?,
                      probe_receive?, probe_address?)
>
```

```
<!ELEMENT tcp_probe (probe_failed?, probe_interval?, probe_retries?,
                     probe_open?, probe_port?)
>

<!ELEMENT udp_probe (probe_failed?, probe_interval?, probe_retries?,
                     probe_receive?, probe_port?)
>

<!ELEMENT smtp_probe (probe_failed?, probe_interval?, probe_retries?,
                      probe_open?, probe_receive?, probe_port?,
                      probe_expect_status*)
>

<!ELEMENT telnet_probe (probe_failed?, probe_interval?, probe_retries?,
                        probe_open?, probe_receive?, probe_port?,
                        probe_expect_status*)
>

<!ELEMENT ftp_probe (probe_failed?, probe_interval?, probe_retries?,
                     probe_open?, probe_receive?, probe_port?,
                     probe_expect_status*)
>

<!ELEMENT script_probe (probe_failed?, probe_interval?, probe_retries?,
                        probe_open?, probe_receive?, probe_port?, script_ref?)
>

<!--
  probe_address must use mode "routed"
-->
<!ELEMENT kalap_udp_probe (probe_failed?, probe_interval?, probe_retries?,
                           probe_receive?, probe_port?, probe_address?,
                           probe_secret?)
>

<!--
  probe_address must use mode "routed"
-->
<!ELEMENT kalap_tcp_probe (probe_failed?, probe_interval?, probe_retries?,
                           probe_open?, probe_receive?, probe_port?,
                           probe_address?, probe_secret?)
>

<!ELEMENT probe (http_probe | dns_probe | icmp_probe | tcp_probe | udp_probe |
                 smtp_probe | telnet_probe | ftp_probe | script_probe |
                 kalap_udp_probe | kalap_tcp_probe)
>
<!ATTLIST probe
  sense (yes | no)                #IMPLIED
  name  CDATA                     #REQUIRED
  type  (http | dns | icmp | tcp | udp |
         smtp | telnet | ftp | script |
         kal-ap-udp | kal-ap-tcp)   #REQUIRED
>

<!--
*************************************************************
  Elements and attributes required for natpool
*************************************************************
-->

<!--
  Global maximum of 255 natpool addresses
-->
```

```
<!ELEMENT natpool EMPTY>
<!ATTLIST natpool
  sense    (yes | no) #IMPLIED
  name     CDATA      #REQUIRED
  first_ip NMTOKEN    #REQUIRED
  last_ip  NMTOKEN    #REQUIRED
  ipmask   NMTOKEN    #REQUIRED
>

<!--
************************************************************
  Elements and attributes required by maps
************************************************************
-->

<!--
  url is a string of length 1 to 200
  method is a string of length 1 to 15 (e.g. GET)
-->
<!ELEMENT url_rule EMPTY>
<!ATTLIST url_rule
  sense  (yes | no) #IMPLIED
  url    CDATA      #REQUIRED
  method CDATA      #IMPLIED
>

<!--
  name is a string of length 1 to 63
  expression is a string of length 1 to 127
-->
<!ELEMENT cookie_rule EMPTY>
<!ATTLIST cookie_rule
  sense      (yes | no) #IMPLIED
  name       CDATA      #REQUIRED
  expression CDATA      #REQUIRED
>

<!--
  name is a string of length 1 to 63
  expression is a string of length 1 to 127
-->
<!ELEMENT header_rule EMPTY>
<!ATTLIST header_rule
  sense      (yes | no) #IMPLIED
  name       CDATA      #REQUIRED
  expression CDATA      #REQUIRED
  type       (match | insert) "match"
>

<!--
  min_code and max_code are between 100 and 599
  threshold is between 1 and 4294967295, no effect for count action
  reset is between 0 and 4294967295 (0 means no reset)
-->
<!ELEMENT retcode_rule EMPTY>
<!ATTLIST retcode_rule
  sense     (yes | no)             #IMPLIED
  min_code  NMTOKEN                #REQUIRED
  max_code  NMTOKEN                #REQUIRED
  action    (count | log | remove) #REQUIRED
  threshold NMTOKEN                #REQUIRED
  reset     NMTOKEN                "0"
>

<!--
  domain is a string of length 1 to 127
-->
```

```
<!ELEMENT dns_rule EMPTY>
<!ATTLIST dns_rule
  sense  (yes | no) #IMPLIED
  domain CDATA      #REQUIRED
>

<!--
  Maximum of 1023 url_rules per map
-->
<!ELEMENT url_map (url_rule*)>
<!ATTLIST url_map
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>

<!--
  Maximum of 5 cookie_rules per map
-->
<!ELEMENT cookie_map (cookie_rule*)>
<!ATTLIST cookie_map
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>

<!--
  Maximum of 5 header_rules per map
-->
<!ELEMENT header_map (header_rule*)>
<!ATTLIST header_map
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>

<!--
  Maximum of 100 retcodes (not ranges) per map
-->
<!ELEMENT retcode_map (retcode_rule*)>
<!ATTLIST retcode_map
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>

<!--
  Maximum of 16 dns_rules per map
-->
<!ELEMENT dns_map (dns_rule*)>
<!ATTLIST dns_map
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>

<!--
************************************************************
  Elements and attributes required for redirect_server
************************************************************
-->

<!--
  value is between 1 and 65535
-->
<!ELEMENT ssl_port EMPTY>
<!ATTLIST ssl_port
  sense (yes | no) #IMPLIED
  value NMTOKEN    #REQUIRED
>

<!--
  string is of length 1 to 127
-->
```

```
<!ELEMENT redirect_relocate EMPTY>
<!ATTLIST redirect_relocate
  sense  (yes | no)  #IMPLIED
  string CDATA       #REQUIRED
  code   (301 | 302) "302"
>

<!--
  string is of length 1 to 127
-->
<!ELEMENT redirect_backup EMPTY>
<!ATTLIST redirect_backup
  sense  (yes | no)  #IMPLIED
  string CDATA       #REQUIRED
  code   (301 | 302) "302"
>

<!ELEMENT redirect_server (ssl_port?, redirect_relocate?, redirect_backup?,
                           inservice?)
>
<!ATTLIST redirect_server
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>

<!--
************************************************************
  Elements and attributes required for named_real_server
************************************************************
-->

<!--
  string is of length 0 to 63
-->
<!ELEMENT location EMPTY>
<!ATTLIST location
  sense  (yes | no) #IMPLIED
  string CDATA      #REQUIRED
>

<!ELEMENT real_address EMPTY>
<!ATTLIST real_address
  sense     (yes | no) #IMPLIED
  ipaddress NMTOKEN    #REQUIRED
>

<!ELEMENT named_real_server (real_address?, location?)>
<!ATTLIST named_real_server
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>

<!--
******************************************************
 Elements and attributes required for state element.
 This is READ ONLY.
******************************************************
-->

<!ELEMENT state EMPTY>
<!ATTLIST state
value (outofservice | operational | standby | outofmemory |
backup_sfarm | init | zombie | closing | estab | synclient |
synserver | synboth | finclient | finserver | failed |
ready_to_test | testing | maxconns | dfp_throttled |
probe_failed | probe_testing | health_failed |
retcode_failed | pending | success |
active | unknown | outofsync | not_connected |
  trying_to_connect | connecting | connected |
  security_error )   #IMPLIED
```

```
>

<!--
*************************************************************
 Elements and attributes required for connections_hits element.
 This is READ ONLY.
*************************************************************
-->

<!ELEMENT connections_hits EMPTY>
<!ATTLIST connections_hits
  value   NMTOKEN   #IMPLIED
>

<!--
*************************************************************
 Elements and attributes required for sticky_entry element.
 This is READ ONLY.
*************************************************************
-->

<!--
id is between 1 and 255
timeout is between 1 and 65535
ipmask required for ip types
cookie is a string of length 1 to 63, req for type=cookie or  cookie_insert
header is a string of length 1 to 63, req for type=header
ip, cookie_hash1, cookie_hash2, ssl_hash1, ssl_hash2,
header_hash1 and  header_hash2 are READ ONLY fields.
-->

<!ELEMENT sticky_entry (sticky_offset?, cookie_secondary?,
                        static_sticky?, real_server?)>

<!ATTLIST sticky_entry
  id      NMTOKEN             #REQUIRED
  timeout NMTOKEN             "1440"
  type    (ip | cookie | ssl |
           ip_src | ip_dest | ip_src_dest |
           cookie_insert | header) #REQUIRED
  ipmask  NMTOKEN             #IMPLIED
  cookie  CDATA               #IMPLIED
  header  CDATA         #IMPLIED
  ip              NMTOKEN           #IMPLIED
  cookie_hash1   NMTOKEN             #IMPLIED
  cookie_hash2   NMTOKEN             #IMPLIED
  ssl_hash1       NMTOKEN             #IMPLIED
  ssl_hash2       NMTOKEN             #IMPLIED
  header_hash1  NMTOKEN             #IMPLIED
  header_hash2  NMTOKEN             #IMPLIED
>



<!--
*************************************************************
  Elements and attributes required for module_connection_stats.
  This is READ ONLY.
*************************************************************
-->

<!ELEMENT module_connection_stats EMPTY>
<!ATTLIST module_connection_stats
  created     NMTOKEN    #IMPLIED
  destroyed   NMTOKEN    #IMPLIED
  current     NMTOKEN    #IMPLIED
  time_out    NMTOKEN    #IMPLIED
  failed      NMTOKEN    #IMPLIED
>
```

```
<!--
************************************************************
  Elements and attributes required for
  server_initiated_connection_stats.
  This is READ ONLY.
************************************************************
-->

<!ELEMENT server_initiated_connection_stats EMPTY>
<!ATTLIST server_initiated_connection_stats
  created      NMTOKEN    #IMPLIED
  current      NMTOKEN    #IMPLIED
  failed       NMTOKEN    #IMPLIED
>

<!--
************************************************************
  Elements and attributes required for l4_lb_stats.
  This is READ ONLY.
************************************************************
-->

<!ELEMENT l4_lb_stats EMPTY>
<!ATTLIST l4_lb_stats
  decisions    NMTOKEN        #IMPLIED
  drops        NMTOKEN        #IMPLIED
>

<!--
************************************************************
  Elements and attributes required for l7_lb_stats.
  This is READ ONLY.
************************************************************
-->

<!ELEMENT l7_lb_stats EMPTY>
<!ATTLIST l7_lb_stats
  decisions                       NMTOKEN        #IMPLIED
  total_drops                     NMTOKEN        #IMPLIED
  max_parse_length_reached_drops  NMTOKEN        #IMPLIED
  cookie_out_of_mem_drops         NMTOKEN        #IMPLIED
  config_version_mismatch_drops   NMTOKEN        #IMPLIED
  bas_ssl2_format_drops           NMTOKEN        #IMPLIED
>

<!--
************************************************************
  Elements and attributes required for l4_l7_lb_stats.
  This is READ ONLY.
************************************************************
-->

<!ELEMENT l4_l7_lb_stats EMPTY >
<!ATTLIST l4_l7_lb_stats
  no_policy_drops          NMTOKEN    #IMPLIED
  no_policy_match_drops    NMTOKEN    #IMPLIED
  no_real_drops            NMTOKEN    #IMPLIED
  acl_denied_drops         NMTOKEN    #IMPLIED
  server_initiated_drops   NMTOKEN    #IMPLIED
>

<!--
************************************************************
  Elements and attributes required for checksum_failures_stats.
  This is READ ONLY.
************************************************************
-->
```

```
<!ELEMENT checksum_failures_stats EMPTY>
<!ATTLIST checksum_failures_stats
  ip     NMTOKEN     #IMPLIED
  tcp    NMTOKEN     #IMPLIED
>

<!--
*************************************************************
  Elements and attributes required for redirect_stats.
  This is READ ONLY.
*************************************************************
-->

<!ELEMENT redirect_stats EMPTY>
<!ATTLIST redirect_stats
  connections    NMTOKEN    #IMPLIED
  drops          NMTOKEN    #IMPLIED
>

<!--
*************************************************************
  Elements and attributes required for ftp_stats.
  This is READ ONLY.
*************************************************************
-->

<!ELEMENT ftp_stats EMPTY>
<!ATTLIST ftp_stats
connections     NMTOKEN     #IMPLIED
>

<!--
*************************************************************
  Elements and attributes required for tx_frame_stats.
  This is READ ONLY.
*************************************************************
-->

<!ELEMENT tx_frame_stats EMPTY>
<!ATTLIST tx_frame_stats
  unicast             NMTOKEN #IMPLIED
  multicast           NMTOKEN #IMPLIED
  broadcast           NMTOKEN #IMPLIED
  underflow_errors    NMTOKEN #IMPLIED
>

<!--
*************************************************************
  Elements and attributes required for rx_frame_stats.
  This is READ ONLY.
*************************************************************
-->

<!ELEMENT rx_frame_stats EMPTY>
<!ATTLIST rx_frame_stats
  unicast          NMTOKEN     #IMPLIED
  multicast        NMTOKEN     #IMPLIED
  broadcast        NMTOKEN     #IMPLIED
  overflow_errors  NMTOKEN     #IMPLIED
  crc_errors       NMTOKEN     #IMPLIED
>


<!--
*************************************************************
  Elements and attributes required for mac_frames_stats.
  This is READ ONLY.
*************************************************************
-->
```

```
<!ELEMENT mac_frames_stats (tx_frame_stats?, rx_frame_stats?)>


<!--
*************************************************************
  Elements and attributes required for stats.
  This is READ ONLY.
*************************************************************
-->

<!ELEMENT stats(module_connection_stats?,
server_initiated_connection_stats?,l4_lb_stats?, l7_lb_stats?,
l4_l7_lb_stats?, checksum_failure_stats?, redirect_stats?,
ftp_stats?, mac_frames_stats?)>


<!--
*************************************************************
  Elements and attributes required for real_server
*************************************************************
-->

<!--
  value is between 0 and 100
-->
<!ELEMENT weight EMPTY>
<!ATTLIST weight
  sense (yes | no) #IMPLIED
  value NMTOKEN    #REQUIRED
>

<!--
  value is between 1 and 4294967295
-->
<!ELEMENT minconns EMPTY>
<!ATTLIST minconns
  sense (yes | no) #IMPLIED
  value NMTOKEN    #REQUIRED
>

<!--
  value is between 2 and 254 (default is 254)
-->
<!ELEMENT load_threshold EMPTY>
<!ATTLIST load_threshold
  sense (yes | no) #IMPLIED
  value NMTOKEN    #REQUIRED
>

<!--
  tag is a string of length 0 to 32
-->
<!ELEMENT real_probe_ref EMPTY>
<!ATTLIST real_probe_ref
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
  tag   CDATA      #IMPLIED
>

<!--
  either ipaddress or named_real_server_ref is required
  port is between 0 and 65535 (0 means no port translation)
-->
<!ELEMENT real_server_backup EMPTY>
<!ATTLIST real_server_backup
  sense                (yes | no) #IMPLIED
  ipaddress            NMTOKEN    #IMPLIED
  named_real_server_ref CDATA     #IMPLIED
  port                 NMTOKEN    "0"
>
```

```
<!--
  either ipaddress or named_real_server_ref is required
  port is between 0 and 65535 (0 means no port translation)
  Global maximum of 4095 real_servers
-->
<!ELEMENT real_server (weight?, minconns?, maxconns?, load_threshold?,
                       real_probe_ref?, real_server_backup?, inservice?,
                       inservice_standby?, serverfarm_ref?, state?,
                       connections_hits?)
>
<!ATTLIST real_server
  sense                 (yes | no) #IMPLIED
  ipaddress             NMTOKEN    #IMPLIED
  named_real_server_ref CDATA      #IMPLIED
  port                  NMTOKEN    "0"
>

<!--
*******************************************************
 Elements and attributes required for num_reals element
 This is READ ONLY.
*******************************************************
-->

<!ELEMENT num_reals EMPTY>
<!ATTLIST num_reals
  value NMTOKEN #IMPLIED
>

<!--
*******************************************************
 Elements and attributes required for num_redirect element
 This is READ ONLY.
*******************************************************
-->

<!ELEMENT num_redirect EMPTY>
<!ATTLIST num_redirect
  value NMTOKEN #IMPLIED
>

<!--
*************************************************************
  Elements and attributes required for serverfarm
*************************************************************
-->

<!ELEMENT retcode_map_ref EMPTY>
<!ATTLIST retcode_map_ref
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>

<!--
  retries is between 0 and 65534
  failed is between 0 and 65535
-->
<!ELEMENT health EMPTY>
<!ATTLIST health
  sense     (yes | no) #IMPLIED
  retries   NMTOKEN    #REQUIRED
  failed    NMTOKEN    #REQUIRED
>

<!ELEMENT failaction EMPTY>
<!ATTLIST failaction
  sense (yes | no)         #IMPLIED
  value (purge | reassign) #REQUIRED
>
```

```
<!ELEMENT probe_ref EMPTY>
<!ATTLIST probe_ref
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>

<!ELEMENT natpool_ref EMPTY>
<!ATTLIST natpool_ref
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>

<!ELEMENT server_nat EMPTY>
<!ATTLIST server_nat
  sense (yes | no) #IMPLIED
>

<!--
  value is between 0 and 65533
-->
<!ELEMENT bind_id EMPTY>
<!ATTLIST bind_id
  sense (yes | no) #IMPLIED
  value NMTOKEN    #REQUIRED
>

<!--
  hash_ip_type and ipmask valid only when value = hash_ip
-->
<!ELEMENT predictor EMPTY>
<!ATTLIST predictor
  sense        (yes | no)                      #IMPLIED
  value        (roundrobin | leastconns |
                hash_ip | hash_url | forward) #REQUIRED
  hash_ip_type (source | destination | both)  "both"
  ipmask       NMTOKEN                        "255.255.255.255"
>

<!ELEMENT dns_predictor EMPTY>
<!ATTLIST dns_predictor
  sense        (yes | no)                      #IMPLIED
  value        (roundrobin | ordered-list |
                leastload | hash_domain |
                hash_ip | hash_ip_domain)     #REQUIRED
>

<!ELEMENT serverfarm (predictor?, natpool_ref?, server_nat?, health?,
                      bind_id?, retcode_map_ref?, failaction?,
redirect_server*, real_server*, probe_ref*,
num_reals?, num_redirect?)
>
<!ATTLIST serverfarm
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>

<!--
  real_server "port" attribute is ignored
-->
<!ELEMENT dns_serverfarm (dns_predictor?, real_server*, num_reals?)>
<!ATTLIST dns_serverfarm
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
  type  (dns-vip | dns-ns) #REQUIRED
>
```

```
<!--
************************************************************
  Elements and attributes required for sticky_group
************************************************************
-->

<!--
  src_ip and dest_ip are necessary for IP-based sticky_groups
  expression is necessary for SSL, cookie, and header-based sticky_groups
  expression is a string of length 0 to 127
-->
<!ELEMENT static_sticky EMPTY>
<!ATTLIST static_sticky
  sense      (yes | no) #IMPLIED
  real_ip    NMTOKEN    #REQUIRED
  expression NMTOKEN    #IMPLIED
  src_ip     NMTOKEN    #IMPLIED
  dest_ip    NMTOKEN    #IMPLIED
>

<!--
  This only applies to cookie and header-based sticky_groups
  offset is between 0 and 3999
  length is between 1 and 4000
-->
<!ELEMENT sticky_offset EMPTY>
<!ATTLIST sticky_offset
  sense (yes | no) #IMPLIED
  offset NMTOKEN   #REQUIRED
  length NMTOKEN   #REQUIRED
>

<!--
  This only applies to cookie-based sticky_groups
  name is a string of length 1 to 63
-->
<!ELEMENT cookie_secondary EMPTY>
<!ATTLIST cookie_secondary
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>

<!--
  id is between 1 and 255
  timeout is between 1 and 65535
  ipmask required for ip types
  cookie is a string of length 1 to 63, req for type=cookie or cookie_insert
  header is a string of length 1 to 63, req for type=header
  "ip" and "num_conns" are READ ONLY fields.
-->
<!ELEMENT sticky_group (sticky_offset?, cookie_secondary?,
                        static_sticky?, real_server?)>
<!ATTLIST sticky_group
  sense   (yes | no)          #IMPLIED
  id      NMTOKEN             #REQUIRED
  timeout NMTOKEN             "1440"
  type    (ip | cookie | ssl |
           ip_src | ip_dest | ip_src_dest |
           cookie_insert | header) #REQUIRED
  ipmask  NMTOKEN             #IMPLIED
  cookie  CDATA               #IMPLIED
  header  CDATA               #IMPLIED
  ip      NMTOKEN             #IMPLIED
  num_conns NMTOKEN       #IMPLIED
>
```

```
<!--
************************************************************
  Elements and attributes required for policy
************************************************************
-->

<!ELEMENT url_map_ref EMPTY>
<!ATTLIST url_map_ref
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>

<!ELEMENT cookie_map_ref EMPTY>
<!ATTLIST cookie_map_ref
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>

<!ELEMENT header_map_ref EMPTY>
<!ATTLIST header_map_ref
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>

<!ELEMENT dns_map_ref EMPTY>
<!ATTLIST dns_map_ref
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>

<!--
  order is between 1 and 3 (corresponds to "primary", "secondary", "tertiary")
  ttl is between 1 and 604800 (default is 20)
  response_count is between 1 and 8 (default is 1)
-->
<!ELEMENT dns_serverfarm_ref EMPTY>
<!ATTLIST dns_serverfarm_ref
  sense (yes | no) #IMPLIED
  order NMTOKEN    #REQUIRED
  name  CDATA      #REQUIRED
  ttl   NMTOKEN    #IMPLIED
  response_count NMTOKEN #IMPLIED
>

<!--
  Reference to an IOS standard IP access list
  Specify either the id (range 1 to 99) or name
  name is a string of length 1 to 200
-->
<!ELEMENT client_group_ref EMPTY>
<!ATTLIST client_group_ref
  sense (yes | no) #IMPLIED
  name CDATA   #IMPLIED
  id   NMTOKEN #IMPLIED
>

<!--
  id is between 1 and 255
-->
<!ELEMENT sticky_group_ref EMPTY>
<!ATTLIST sticky_group_ref
  sense (yes | no) #IMPLIED
  id    NMTOKEN    #REQUIRED
>

<!--
  value is between 0 and 63
-->
```

```
<!ELEMENT dscp EMPTY>
<!ATTLIST dscp
  sense (yes | no) #IMPLIED
  value NMTOKEN    #REQUIRED
>

<!ELEMENT policy (serverfarm_ref?, client_group_ref?, sticky_group_ref?,
                  reverse_sticky?, dscp?, url_map_ref?, cookie_map_ref?,
                  header_map_ref?)
>
<!ATTLIST policy
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>

<!--
  Maximum of 3 dns_serverfarm_refs per dns_policy (one for each order)
-->
<!ELEMENT dns_policy (dns_serverfarm_ref*, client_group_ref?, dns_map_ref?)>
<!ATTLIST dns_policy
  sense (yes | no) #IMPLIED
  name CDATA       #REQUIRED
>

<!--
*******************************************************
 Elements and attributes required for num_conns element.
 This is READ ONLY.
*******************************************************
-->

<!ELEMENT num_conns EMPTY>
<!ATTLIST num_conns
  value NMTOKEN #IMPLIED
>

<!--
*******************************************************
 Elements and attributes required for arp element
 This is READ ONLY.
*******************************************************
-->

<!ELEMENT arp EMPTY>
<!ATTLIST arp
  macaddress NMTOKEN #IMPLIED
  ipaddress NMTOKEN #IMPLIED
  vlan_id  NMTOKEN #IMPLIED
  type  NMTOKEN #IMPLIED
  status   NMTOKEN #IMPLIED
  misses           NMTOKEN     #IMPLIED
>

<!--
*************************************************************
 Elements and attributes required for gslb_client_group element
 This is READ ONLY.
*************************************************************
-->

<!ELEMENT gslb_client_group EMPTY>
<!ATTLIST gslb_client_group
  acl_name     CDATA   #IMPLIED
  acl_id       NMTOKEN    #IMPLIED
  cl_grp_hit_countNMTOKEN    #IMPLIED
>
```

```
<!--
************************************************************
 Elements and attributes required for gslb_dns element
 This is READ ONLY.
************************************************************
-->

<!ELEMENT gslb_dns EMPTY>
<!ATTLIST gslb_dns
  dns_queries_rcvdNMTOKEN    #IMPLIED
  dns_host_addr_query_rcvdNMTOKEN    #IMPLIED
  dns_responses_sentNMTOKEN    #IMPLIED
  dns_responses_no_errNMTOKEN    #IMPLIED
  dns_responses_errNMTOKEN    #IMPLIED
  dns_queries_unmatchedNMTOKEN    #IMPLIED
  dns_drops    NMTOKEN    #IMPLIED
  dns_ns_fwd_sentNMTOKEN    #IMPLIED
  dns_ns_fwd_resp_rcvdNMTOKEN    #IMPLIED
  dns_curr_req_per_secNMTOKEN    #IMPLIED
  dns_peak_req_per_secNMTOKEN    #IMPLIED
>

<!--
*******************************************************
 Elements and attributes required for gslb_probe element
 This is READ ONLY.
*******************************************************
-->

<!ELEMENT gslb_probe EMPTY>
<!ATTLIST gslb_probe
  probe_type   CDATA    #IMPLIED
  probe_successes      NMTOKEN    #IMPLIED
  probe_failures              NMTOKEN     #IMPLIED
  probe_pkts_sent     NMTOKEN    #IMPLIED
  probe_pkts_rcvd NMTOKEN    #IMPLIED
>

<!--
*******************************************************
 Elements and attributes required for gslb element
 This is READ ONLY.
*******************************************************
-->

<!ELEMENT gslb (gslb_probe*, gslb_dns?, gslb_client_group?)>

<!--
*******************************************************
 Elements and attributes required for probe_real element
 This is READ ONLY.
*******************************************************
-->

<!ELEMENT probe_real (state?)>
<!ATTLIST probe_real
  real_ip   NMTOKEN    #IMPLIED
  port      NMTOKEN    #IMPLIED
  probe_name     CDATA    #IMPLIED
  vserver_name CDATA   #IMPLIED
  sfarm_name   CDATA   #IMPLIED
  policy_name  CDATA   #IMPLIED
  status       CDATA   #IMPLIED
  current      CDATA   #IMPLIED
  successes    CDATA   #IMPLIED
  last_success CDATA   #IMPLIED
  failures     CDATA   #IMPLIED
  last_failure CDATA   #IMPLIED
>
```

```
<!--
*******************************************************
 Elements and attributes required for downloaded_config
 element . This is READ ONLY.
*******************************************************
-->

<!ELEMENT downloaded_config (state?)>

<!--
*******************************************************
 Elements and attributes required for status element
 This is READ ONLY.
*******************************************************
-->

<!ELEMENT status (downloaded_config?)>
<!ATTLIST status
  slot NMTOKEN #IMPLIED
>

<!--
*******************************************************
 Elements and attributes required for conns element
 This is READ ONLY.
*******************************************************
-->

<!ELEMENT conns (state?)>
<!ATTLIST conns
  type  (in | out) #IMPLIED
  protocol   NMTOKEN     #IMPLIED
  vlan       NMTOKEN    #IMPLIED
  src_port       NMTOKEN    #IMPLIED
  src_ip       NMTOKEN    #IMPLIED
  dest_port      NMTOKEN    #IMPLIED
  dest_ip      NMTOKEN    #IMPLIED
  vname NMTOKEN#IMPLIED
  ftp         NMTOKEN     #IMPLIED
  csrp  NMTOKEN     #IMPLIED
 >

<!--
*******************************************************
 Elements and attributes required for memory element
 This is READ ONLY.
*******************************************************
-->

<!ELEMENT memory EMPTY>
<!ATTLIST memory
  vserver_name                CDATA      #IMPLIED
  error_string                NMTOKEN    #IMPLIED
  total_bytes                 NMTOKEN    #IMPLIED
  first_type_memory           NMTOKEN    #IMPLIED
  second_type_memory         NMTOKEN    #IMPLIED
  total_first_type_memory     NMTOKEN    #IMPLIED
  max_first_type_memory       NMTOKEN    #IMPLIED
  total_second_type_memory    NMTOKEN    #IMPLIED
  max_second_type_memory      NMTOKEN    #IMPLIED
 >

<!--
*******************************************************
 Elements and attributes required for pvlan element
 This is READ ONLY.
*******************************************************
-->
```

```
<!ELEMENT pvlan EMPTY>
<!ATTLIST pvlan
  primary         NMTOKEN    #IMPLIED
  secondary       NMTOKEN    #IMPLIED
 >

<!--
*******************************************************
 Elements and attributes required for clear_conns element
 *******************************************************
-->

<!ELEMENT clear_conns (vserver?, real_server?)>
<!ATTLIST clear_conns
  type    (vserver_name | real_ip)     #IMPLIED
>

<!--
************************************************************
  Elements and attributes required for vserver
************************************************************
-->

<!--
  protocol is between 0 and 255 (0 = any, 1 = icmp, 6 = tcp, 17 = udp)
  port is between 0 and 65535 (0 means any)
  ftp and termination service valid only for tcp protocol
  rtsp service valid for tcp and udp protocol
  per-packet service valid only for non-tcp protocols
-->
<!ELEMENT virtual EMPTY>
<!ATTLIST virtual
  sense     (yes | no)          #IMPLIED
  ipaddress NMTOKEN             #REQUIRED
  ipmask    NMTOKEN             "255.255.255.255"
  protocol  NMTOKEN             #REQUIRED
  port      NMTOKEN             #REQUIRED
  service   (none | ftp | rtsp |
             termination | per-packet) "none"
>

<!ELEMENT client EMPTY>
<!ATTLIST client
  sense     (yes | no) #IMPLIED
  ipaddress NMTOKEN    #REQUIRED
  ipmask    NMTOKEN    "255.255.255.255"
  exclude   (yes | no) "no"
>

<!--
  timeout is between 1 and 65535
  group is between 0 and 255 (if nonzero, refers to an ip sticky_group)
-->
<!ELEMENT sticky EMPTY>
<!ATTLIST sticky
  sense   (yes | no) #IMPLIED
  timeout NMTOKEN    #REQUIRED
  group   NMTOKEN    "0"
  ipmask  NMTOKEN    "255.255.255.255"
>

<!ELEMENT policy_ref EMPTY>
<!ATTLIST policy_ref
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>
```

```
<!ELEMENT dns_policy_ref EMPTY>
<!ATTLIST dns_policy_ref
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>

<!--
  begin and end are strings, 0-length ok
  total length of begin and end should not exceed 200
-->
<!ELEMENT url_hash EMPTY>
<!ATTLIST url_hash
  sense (yes | no) #IMPLIED
  begin CDATA      #REQUIRED
  end   CDATA      #REQUIRED
>

<!--
  value is between 2 and 4094
-->
<!ELEMENT vlan_id EMPTY>
<!ATTLIST vlan_id
  sense (yes | no) #IMPLIED
  value NMTOKEN    #REQUIRED
>

<!--
  value is between 2 and 65535
-->
<!ELEMENT idle EMPTY>
<!ATTLIST idle
  sense (yes | no) #IMPLIED
  value NMTOKEN    #REQUIRED
>

<!--
  value is between 1 and 65535
-->
<!ELEMENT pending EMPTY>
<!ATTLIST pending
  sense (yes | no) #IMPLIED
  value NMTOKEN    #REQUIRED
>

<!ELEMENT replicate_csrp EMPTY>
<!ATTLIST replicate_csrp
  sense (yes | no)            #IMPLIED
  value (sticky | connection) #REQUIRED
>

<!ELEMENT advertise EMPTY>
<!ATTLIST advertise
  sense (yes | no)        #IMPLIED
  value (always | active) #REQUIRED
>

<!ELEMENT persistent EMPTY>
<!ATTLIST persistent
  sense (yes | no) #IMPLIED
>

<!--
  value is between 1 and 4000
-->
<!ELEMENT parse_length EMPTY>
<!ATTLIST parse_length
  sense (yes | no) #IMPLIED
  value NMTOKEN    #REQUIRED
>
```

```
<!--
  string is of length 1 to 127
-->
<!ELEMENT domain EMPTY>
<!ATTLIST domain
  sense  (yes | no) #IMPLIED
  string CDATA      #REQUIRED
>

<!ELEMENT unidirectional EMPTY>
<!ATTLIST unidirectional
  sense (yes | no | default) #IMPLIED
>

<!ELEMENT owner_ref EMPTY>
<!ATTLIST owner_ref
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>

<!--
  offset is between 0 and 3999
  length is between 1 and 4000
-->
<!ELEMENT ssl_sticky_offset EMPTY>
<!ATTLIST ssl_sticky_offset
  sense (yes | no) #IMPLIED
  offset NMTOKEN   #REQUIRED
  length NMTOKEN   #REQUIRED
>

<!--
  Maximum of 1023 domains per vserver
  Default idle is 3600
  Default pending is 30
-->
<!ELEMENT vserver (virtual?, vlan_id?, unidirectional?, owner_ref?,
                maxconns?, ssl_sticky_offset?, idle?, pending?,
                replicate_csrp?, advertise?, persistent?, parse_length?,
              inservice?, url_hash?, policy_ref*, domain*,
                serverfarm_ref?, sticky?, reverse_sticky?, client*,
                num_conns?, state?)
>
<!ATTLIST vserver
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>

<!ELEMENT dns_vserver (inservice?, dns_policy_ref*)>
<!ATTLIST dns_vserver
  sense (yes | no) #IMPLIED
  name  CDATA      #REQUIRED
>

<!--
************************************************************
  Elements and attributes required for dfp.
  ip, password, anonymous_client, client_ip, client_port,
  Interval, messages, last_connect, security_erss,
  last_security_err are READ ONLY.
************************************************************
-->

<!--
  port is between 1 and 65535
-->
```

```
<!ELEMENT dfp_manager EMPTY>
<!ATTLIST dfp_manager
  sense            (yes | no)     #IMPLIED
  port              NMTOKEN       #IMPLIED
  ip                NMTOKEN       #IMPLIED
  password          NMTOKEN       #IMPLIED
  anonymous_client  NMTOKEN       #IMPLIED
  client_ip         NMTOKEN       #IMPLIED
  client_port       NMTOKEN       #IMPLIED
  interval          NMTOKEN       #IMPLIED
  messages          NMTOKEN       #IMPLIED
  last_connect      NMTOKEN       #IMPLIED
  security_erss     NMTOKEN       #IMPLIED
  last_security_err NMTOKEN       #IMPLIED
>

<!--
  port is between 1 and 65535
  timeout is between 0 and 65535
  retry is between 0 and 65535 (must specify timeout)
  interval is between 1 and 65535 (must specify retry).
  retries, security_errors, last_msg_recv are READ ONLY.
-->
<!ELEMENT dfp_agent (state?)>
<!ATTLIST dfp_agent
  sense        (yes | no)      #IMPLIED
  ipaddress        NMTOKEN        #IMPLIED
  port             NMTOKEN       #IMPLIED
  timeout          NMTOKEN       "0"
  retry            NMTOKEN       "0"
  interval         NMTOKEN       "180"
  retries          NMTOKEN       #IMPLIED
  security_errors  NMTOKEN       #IMPLIED
  last_msg_recv    NMTOKEN       #IMPLIED
>

<!--
************************************************************
  Elements and attributes required for dfp_weight.
  This is READ ONLY.
************************************************************
-->
<!ELEMENT dfp_weight EMPTY>
<!ATTLIST dfp_weight
  ip         NMTOKEN       #IMPLIED
  protocol   CDATA         #IMPLIED
  bind_id    NMTOKEN       #IMPLIED
  weight     NMTOKEN       #IMPLIED
  set_time   NMTOKEN       #IMPLIED
  agent_ip   NMTOKEN       #IMPLIED
  agent_port NMTOKEN       #IMPLIED
>

<!--
************************************************************
  Elements and attributes required for dfp_password.
  This is READ ONLY.
************************************************************
-->
<!ELEMENT dfp_password EMPTY>
<!ATTLIST dfp_password
  password CDATA      #IMPLIED
  pending  CDATA      #IMPLIED
  timeout  NMTOKEN    #IMPLIED
>
```

```
<!--
  password is a string of length 1 to 64
  timeout is between 0 and 65535
-->
<!ELEMENT dfp (dfp_password?, dfp_manager?, dfp_agent*, dfp_weight?)>
<!ATTLIST dfp
  sense    (yes | no) #IMPLIED
  password CDATA      #IMPLIED
  timeout  NMTOKEN    "180"
>



<!--
*************************************************************
  Elements and attributes required for udp_capp
*************************************************************
-->

<!--
  secret is a string of length 1 to 32.
  trans_frames, trans_bytes, trans_errors, recv_frames,
  recv_bytes, recv_errors are all READ ONLY.
-->
<!ELEMENT capp_options EMPTY>
<!ATTLIST capp_options
  sense      (yes | no) #IMPLIED
  ipaddress  NMTOKEN    #IMPLIED
  encryption (md5)      "md5"
  secret     CDATA      #IMPLIED
  trans_frames NMTOKEN    #IMPLIED
  trans_bytes  NMTOKEN    #IMPLIED
  trans_errors NMTOKEN    #IMPLIED
  recv_frames  NMTOKEN    #IMPLIED
  recv_bytes   NMTOKEN    #IMPLIED
  recv_errors NMTOKEN    #IMPLIED
>

<!--
  value is between 1 and 65535
-->
<!ELEMENT capp_port EMPTY>
<!ATTLIST capp_port
  sense (yes | no) #IMPLIED
  value NMTOKEN    #REQUIRED
>

<!ELEMENT capp_secure EMPTY>
<!ATTLIST capp_secure
  sense (yes | no) #IMPLIED
>

<!--
  Maximum of 16 capp_options
  Default capp_port is 5002
-->
<!ELEMENT udp_capp (capp_port?, capp_secure?, capp_options*)>
<!ATTLIST udp_capp
  sense (yes | no) #IMPLIED
>

<!--
*************************************************************
  Elements and attributes required for ft
*************************************************************
-->
```

```
<!ELEMENT ft_preempt EMPTY>
<!ATTLIST ft_preempt
  sense (yes | no) #IMPLIED
>

<!--
  value is between 1 and 254
-->
<!ELEMENT ft_priority EMPTY>
<!ATTLIST ft_priority
  sense (yes | no) #IMPLIED
  value NMTOKEN    #REQUIRED
>

<!--
  value is between 1 and 65535
-->
<!ELEMENT ft_failover EMPTY>
<!ATTLIST ft_failover
  sense (yes | no) #IMPLIED
  value NMTOKEN #REQUIRED
>

<!--
  value is between 1 and 65535
-->
<!ELEMENT ft_heartbeat EMPTY>
<!ATTLIST ft_heartbeat
  sense (yes | no) #IMPLIED
  value NMTOKEN    #REQUIRED
>

<!--
  group is between 1 and 254
  vlan_id is between 2 and 4094, and must *not* match id of
    existing client or server vlan configured for csm_module
  Default ft_preempt is off
  Default ft_priority is 10
  Default ft_failover is 3
  Default ft_heartbeat is 1
-->
<!ELEMENT ft (state?, ft_preempt?, ft_priority?, ft_failover?, ft_heartbeat?)>
<!ATTLIST ft
  sense   (yes | no) #IMPLIED
  group   NMTOKEN    #REQUIRED
  vlan_id NMTOKEN    #REQUIRED
>

<!--
************************************************************
  Elements and attributes required for static_nat
  num_servers is READ ONLY field.
************************************************************
-->

<!ELEMENT static_real EMPTY>
<!ATTLIST static_real
  sense      (yes | no) #IMPLIED
  ipaddress   NMTOKEN    #REQUIRED
  ipmask      NMTOKEN    "255.255.255.255"
  num_servers NMTOKEN  #IMPLIED
>

<!--
  ipaddress is required for type=ip
  Global maximum of 16383 static_reals
-->
```

```
<!ELEMENT static_nat (static_real*)>
<!ATTLIST static_nat
  sense     (yes | no)            #IMPLIED
  type      (drop | ip | virtual) #REQUIRED
  ipaddress NMTOKEN               #IMPLIED
>

<!--
************************************************************
  Elements and attributes required for static_arp
************************************************************
-->

<!--
  macaddress has the form "hhhh.hhhh.hhhh", where h is a hex digit
  vlan_id is between 2 and 4094
-->
<!ELEMENT static_arp EMPTY>
<!ATTLIST static_arp
  sense      (yes | no) #IMPLIED
  ipaddress  NMTOKEN    #REQUIRED
  macaddress NMTOKEN    #REQUIRED
  vlan_id    NMTOKEN    #REQUIRED
>

<!--
*****************************************************
 Elements and attributes required by show_vserver element
*****************************************************
-->

<!ELEMENT show_vserver EMPTY>
<!ATTLIST show_vserver
  redirect   (yes | no)  #IMPLIED
  name       CDATA   #IMPLIED
  config     (yes | no)  #IMPLIED
  detail     (yes | no)  #IMPLIED
  owner_name CDATA #IMPLIED
>
<!--
*****************************************************
 Elements and attributes required by show_serverfarm element
*****************************************************
-->

<!ELEMENT show_serverfarm EMPTY>
<!ATTLIST show_serverfarm
  name CDATA   #IMPLIED
  detail (yes | no) #IMPLIED
>

<!--
*****************************************************
 Elements and attributes required by show_real element
*****************************************************
-->

<!ELEMENT show_real EMPTY>
<!ATTLIST show_real
  location  CDATA   #IMPLIED
  retcode   (yes | no)   #IMPLIED
  sfarm     CDATA   #IMPLIED
  detail    (yes | no)   #IMPLIED
>
<!--
*****************************************************
 Elements and attributes required by show_vlan element
*****************************************************
-->
```

```
<!--
id is between 2 and 4094
-->
<!ELEMENT show_vlan EMPTY>
<!ATTLIST show_vlan
  detail ( yes | no)   #IMPLIED
  type   ( client | server | ft )#IMPLIED
  id  NMTOKEN      #IMPLIED
>

<!--
*****************************************************
 Elements and attributes required by show_conns element
*****************************************************
-->

<!ELEMENT show_conns EMPTY>
<!ATTLIST show_conns
  detail    ( yes | no)#IMPLIED
  client_ipNMTOKEN#IMPLIED
  vserver CDATA   #IMPLIED
>
<!--
*****************************************************
 Elements and attributes required by show_policy element
*****************************************************
-->

<!ELEMENT show_policy EMPTY>
<!ATTLIST show_policy
  name     CDATA   #IMPLIED
>
<!--
*****************************************************
 Elements and attributes required by show_map element
*****************************************************
-->

<!ELEMENT show_map EMPTY>
<!ATTLIST show_map
  name CDATA           #IMPLIED
  detail ( yes | no)   #IMPLIED
  type (header | cookie | retcode | url) #IMPLIED
>
<!--
*****************************************************
 Elements and attributes required by show_ft element
*****************************************************
-->

<!ELEMENT show_ft EMPTY>
<!ATTLIST show_ft
  detail ( yes | no)   #IMPLIED
>
<!--
*****************************************************
 Elements and attributes required by show_status element
*****************************************************
-->

<!ELEMENT show_status EMPTY>
<!ATTLIST show_status
  config     ( yes | no)   #IMPLIED
>
```

```
<!--
*****************************************************
 Elements and attributes required by show_probe element
*****************************************************
-->

<!ELEMENT show_probe EMPTY>
<!ATTLIST show_probe
  detail ( yes | no)   #IMPLIED
  nameCDATA            #IMPLIED
  type      (http | dns | icmp | tcp | udp |
            smtp | telnet | ftp | script |
                        kal-ap-udp | kal-ap-tcp | script )   #IMPLIED
>

<!--
***********************************************************
 Elements and attributes required by show_probe_real element
***********************************************************
-->

<!ELEMENT show_probe_real EMPTY>
<!ATTLIST show_probe_real
  real_ip   NMTOKEN#IMPLIED
>

<!--
*****************************************************
 Elements and attributes required by show_sticky element
*****************************************************
-->

<!--
group_no is between 1 and 255
-->

<!ELEMENT show_sticky EMPTY>
<!ATTLIST show_sticky
  client_ip NMTOKEN#IMPLIED
  config( yes | no)   #IMPLIED
  group_idNMTOKEN#IMPLIED
  cookie    CDATA#IMPLIED
  ssl       CDATA#IMPLIED
>

<!--
*****************************************************
 Elements and attributes required by show_natpool element
*****************************************************
-->

<!ELEMENT show_natpool EMPTY>
<!ATTLIST show_natpool
  detail ( yes | no)   #IMPLIED
  nameCDATA #IMPLIED
>
<!--
*****************************************************
 Elements and attributes required by show_arp element
*****************************************************
-->

<!ELEMENT show_arp EMPTY>
```

```
<!--
*****************************************************
 Elements and attributes required by show_variable element
*****************************************************
-->

<!ELEMENT show_variable EMPTY>
<!ATTLIST show_variable
  detail ( yes | no)   #IMPLIED
  name CDATA   #IMPLIED
>

<!--
*******************************************************
 Elements and attributes required by show_owner element
*******************************************************
-->
<!ELEMENT show_owner EMPTY>
<!ATTLIST show_owner
  detail ( yes | no)   #IMPLIED
  nameCDATA   #REQUIRED
>

<!--
*********************************************************
 Elements and attributes required by show_xml_stats element
*********************************************************
-->
<!ELEMENT show_xml_stats EMPTY>
<!ATTLIST show_xml_stats
  detail ( yes | no)#IMPLIED
>

<!--

*****************************************************
 Elements and attributes required by show_memory element
*****************************************************
-->

<!ELEMENT show_memory EMPTY>
<!ATTLIST show_memory
  vserver_name CDATA   #IMPLIED
>

<!--
*****************************************************
 Elements and attributes required by show_pvlan element
*****************************************************
-->
<!ELEMENT show_pvlan EMPTY>

<!--
*****************************************************
 Elements and attributes required by show_stats element
*****************************************************
-->

<!ELEMENT show_stats EMPTY>

<!--
*****************************************************
 Elements and attributes required by show_capp element
*****************************************************
-->
```

```
<!ELEMENT show_capp EMPTY>
<!ATTLIST show_capp
  udp ( yes | no )   #IMPLIED
  detail    ( yes | no )#IMPLIED
>

<!--
*****************************************************
 Elements and attributes required by show_gslb element
*****************************************************
-->

<!ELEMENT show_gslb EMPTY>
<!ATTLIST show_gslb
  probe ( yes | no )   #IMPLIED
  detail    ( yes | no )#IMPLIED
  cl_group  ( yes | no )            #IMPLIED
>

<!--
*****************************************************
 Elements and attributes required by show_dfp element
*****************************************************
-->

<!ELEMENT show_dfp EMPTY>
<!ATTLIST show_dfp
  detail    ( yes | no )#IMPLIED
  agent     ( yes | no )            #IMPLIED
  weights   ( yes | no )            #IMPLIED
  manager   ( yes | no )            #IMPLIED
  ipaddress  NMTOKEN                #IMPLIED
  port       NMTOKEN                #IMPLIED
>

<!--
*******************************************************
 Elements and attributes required by show_static_nat element
*******************************************************
-->

<!ELEMENT show_static_nat EMPTY>
<!ATTLIST show_static_nat
  server    ( yes | no )            #IMPLIED
  nat_type  ( drop, vip, nat_ip )   #IMPLIED
  server_ip  NMTOKEN                #IMPLIED
  ip         NMTOKEN                #IMPLIED
 >

<!--
*******************************************************
 Elements and attributes required by show_script element
*******************************************************
-->

<!ELEMENT show_script EMPTY>
<!ATTLIST show_script
  code    ( yes | no )            #IMPLIED
  file    CDATA                   #IMPLIED
  name    CDATA                   #IMPLIED
>

<!--
*******************************************************
 Elements and attributes required by show_script_task element
*******************************************************
-->
```

```
<!ELEMENT show_script_task EMPTY>
<!ATTLIST show_script_task
  detail    ( yes | no )            #IMPLIED
  index     CDATA                   #IMPLIED
  name      CDATA                   #IMPLIED
>

<!--
************************************************************
  root definition for csm_module
************************************************************
-->

<!--
  slot is between 1 and MAXSLOT (depends on chassis)
  Maximum of 4095 probes
  Maximum of 1023 url_maps
  Maximum of 1023 cookie_maps
  Maximum of 1023 header_maps
  Maximum of 1023 retcode_maps
  Maximum of 1023 dns_maps
  Maximum of 4095 serverfarms and dns_serverfarms
  Maximum of 255 sticky_groups (including those id=0 groups created
                                implicitly for vservers)
  Maximum of 4000 vservers and dns_vservers
  Maximum of 255 owners
  Maximum of 16383 static_arp entries
-->
<!ELEMENT csm_module (env_variable*, owner*, vlan*, script_file*, script_task*,
                      probe*, natpool*, url_map*, cookie_map*, header_map*,
                      retcode_map*, dns_map*, named_real_server*,
                      serverfarm*, dns_serverfarm*, sticky_group*,
                      policy*, dns_policy*, vserver*, dns_vserver*,
                      dfp?, udp_capp?, ft?, static_nat*, static_arp*)
>
<!ATTLIST csm_module
  sense (yes | no) #IMPLIED
  slot  NMTOKEN    #REQUIRED
>

<!--
************************************************************
  actions
************************************************************
-->

<!--
  error_tolerance is a 32-bit value, specified
    in hex or decimal, which acts as a bitmask
    for specifying which error types should be
    ignored.  See valid error types below.  Default is 0x0048.
  dtd_version is a string that specifies the set of
    configurable CSM features, and should match the CSM version
    specified at the top of this DTD.  Default is "2.2".
    Note that if the version is higher than the CSM can
    handle, an error may be returned.  In most cases,
    the CSM will do its best to interpret the document,
    even if dtd_version is missing or higher than expected.
-->
<!ELEMENT config (csm_module)>
<!ATTLIST config
  error_tolerance NMTOKEN #IMPLIED
  dtd_version     NMTOKEN #IMPLIED

<!--
**************
In case of error, the response document will include an "error" child element
in the offending element.  The error element takes the form:
```

```
<!ELEMENT error EMPTY>
<!ATTLIST error
  code NMTOKEN #REQUIRED
>
The body of the error element is a description string.
Attribute "code" is a hex value representing a mask of possible error codes:
    XML_ERR_INTERNAL          = 0x0001 /* internal memory or coding error */
    XML_ERR_COMM_FAILURE      = 0x0002 /* communication failure */
    XML_ERR_WELLFORMEDNESS    = 0x0004 /* not a wellformed XML document */
    XML_ERR_ATTR_UNRECOGNIZED = 0x0008 /* found an unrecognized attribute */
    XML_ERR_ATTR_INVALID      = 0x0010 /* found invalid value in attribute */
    XML_ERR_ATTR_MISSING      = 0x0020 /* required attribute missing */
    XML_ERR_ELEM_UNRECOGNIZED = 0x0040 /* found an unrecognized element */
    XML_ERR_ELEM_INVALID      = 0x0080 /* found invalid element */
    XML_ERR_ELEM_MISSING      = 0x0100 /* required element missing */
    XML_ERR_ELEM_CONTEXT      = 0x0200 /* valid element found in wrong place */
    XML_ERR_IOS_PARSER        = 0x0400 /* IOS unable to parse command */
    XML_ERR_IOS_MODULE_IN_USE = 0x0800 /* Another user is configuring CSM */
    XML_ERR_IOS_WRONG_MODULE  = 0x1000 /* Tried to configure unavailable CSM */
    XML_ERR_IOS_CONFIG        = 0x2000 /* IOS configuration error */
**************
-->
```

C

D

E

F

H

I

M

N

P

R

こ

さ

し

す

せ

そ

た

て

と

は

ふ

へ

ほ

ま

め

も

り

る

ろ