シングルサインオン導入プロジェクトテンプレート
方式設計書

Ver 1.0.3 （2013/09/02）

（株）野村総合研究所
オープンソースソリューション推進室

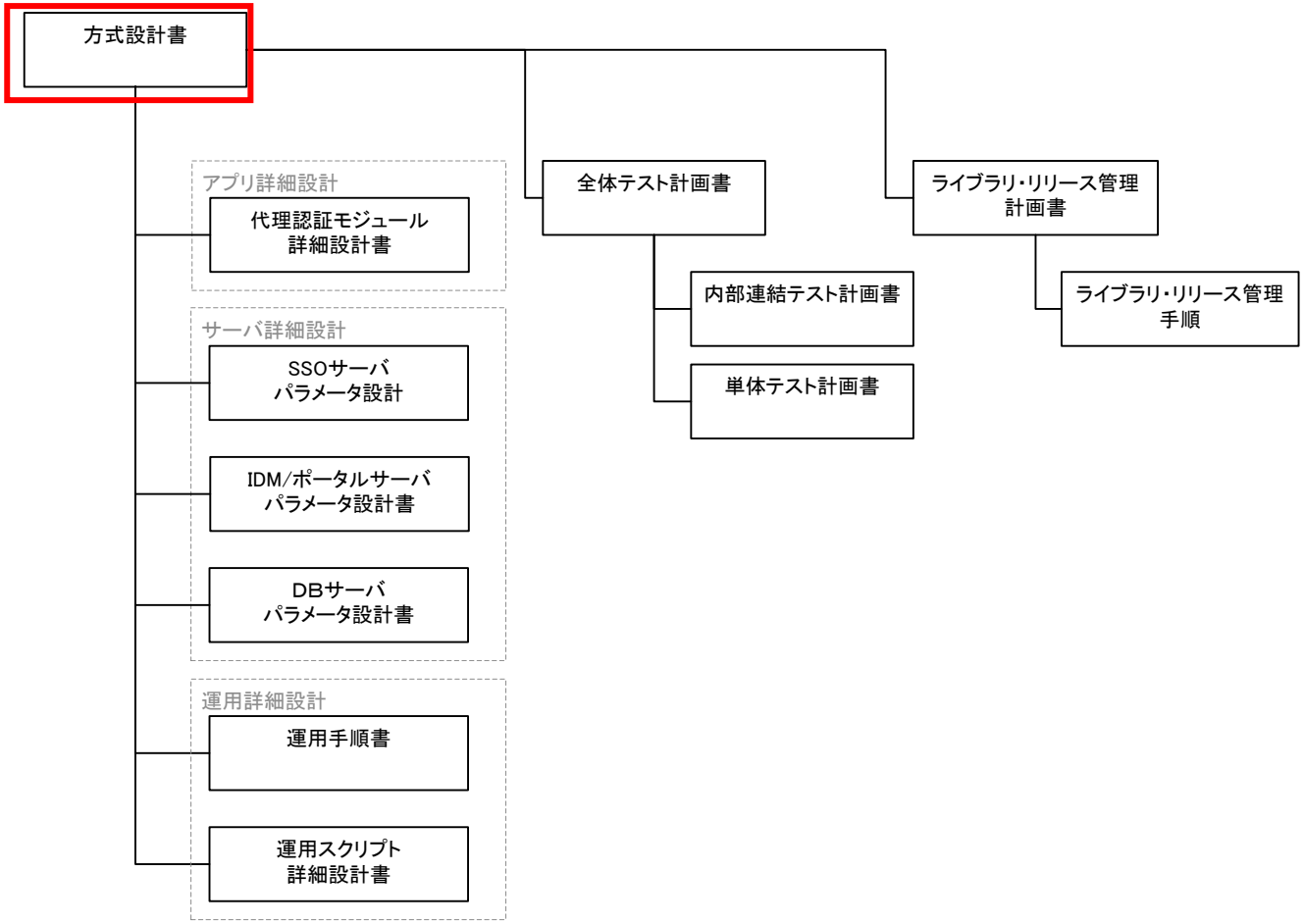
変更履歴

版	日付	担当者	変更内容	備考
1.0	2013/03/22	渡部	初版作成	
1.0.1	2013/04/27	和田	端末要件の表に、〇を付けた	
1.0.2	2013/06/14	渡部	要件定義の章に、記載例を追記 要件定義の章に書いてあった制約事項が、システムの制約事項であり不適切であったため、別の章に移動した	
1.0.3	2013/09/02	和田	タイトルを修正 IDM ポータルの監査ログ種類に監査ログ、サービス監査ログを追記	

目次

1. はじめに

1.1. 本ドキュメントについて



1.2. 本ドキュメントの構成

2章にてシステムの要件を定義し、3章以降で要件を満たすシステムの実現方式を記載する。

システムの要件と実現方式記載箇所の対応一覧

要件			実現方式
機能要件(2.6章)		→	処理方式(4章)
非機能要件(2.7章)	運用要件	→	運用設計(5章)
	セキュリティ要件	→	セキュリティ設計(6章)
	耐障害性要件	→	耐障害設計(7章)
	性能要件	→	性能設計(サイジング)(8章)
	拡張要件	→	拡張設計(9章)
	移行要件	→	移行設計(**エラー! 参照元が見つかりません。**章)
	端末要件	→	端末設計(**エラー! 参照元が見つかりません。**章)
	維持管理・サポート要件	→	維持管理・サポート方針(12章)
	開発要件	→	開発方針(13章)

2. 要件定義

この章は、全工程で作成された要件定義書の内容を整理して記載する。

2.1. プロジェクトの目的／方針

2.1.1. プロジェクトの目的

- ID 管理業務のコストを○○万円削減する。
- 社内システムの共用アカウントを廃止し、○○セキュリティガイドラインを守れるようにする。
- D:○月△日までにリリースする

2.1.2. プロジェクトの目標

- Q:リリース後の障害によるサービス停止時間　1ヶ月中 1 時間以内にする
- C:予算○○万円で目的を達成する
- D:○月△日までにリリースする

2.1.3. プロジェクトの方針

- A システムと B システムをシングルサインオンすることにより、ID 管理業務の工数削減および作業員の作業効率向上を行い、コスト○○万円削減を実現する。

- シングルサインオンのテンプレートである OpenStandia/SSO&IDM テンプレートを用いることにより、開発工数を抑えて、予算〇〇万円を達成させるとともに、工期を短縮し、〇月△日のリリースを守る。

2.2. スコープ

- A システム、B システムのシングルサインオンのみをスコープとして、C システム、D システムは次プロジェクトにて検討する。
- （顧客のシステム全体像を書き、その中でスコープとする部分を明記する）

2.3. 前提

- 貴社にて、A システムの改修を〇月△日までに実施済みである事。

2.4. 制約事項

- 社員の端末は、直接インターネットには接続できず、認証プロキシ経由となる。

2.5. スケジュール

- 2012 年 12 月末までにリリース完了
- （ガントチャートを記載する事）

2.6. 機能要件

2.6.1. アクター一覧

項番	アクター	内容	備考
1	A システムユーザ	A システムを利用できるユーザ	
2	A システム管理者	Aシステムの管理画面にアクセスできるユーザ	
3	B システムユーザ	B システムを利用できるユーザ	
4	B システム管理者	B システムの管理画面にアクセスできるユーザ	

5	ID管理者	ポータルの管理画面にアクセスし、ユーザ・ロールを定義できる人	
6	システム管理者	システム全体を管理できる人	

2.6.2. 機能一覧

項番	大分類	中分類	小分類	内容	備考
1	シングルサインオン	ログイン	連携先システムA（リバースプロキシ型）	ログイン画面を表示し、システムで共通の ID とパスワードにより連携先システムAにログインできる。	
2			連携先システムB（リバプロ＋代理認証）	ログイン画面を表示し、システムで共通の ID とパスワードにより連携先システム B にログインできる。	
3			ポータルログイン（＝エージェント型）	ログイン画面を表示し、システムで共通の ID とパスワードによりポータルにログインできる。	
4			連携先システム D（agent 型）		工数がかかるため、対応しない
5			連携先システムE（SAML 型）		工数がかかるため、対応しない
6		ログイン中チェック（クッキーチェック）	連携先システムA	ログイン中である場合に再度ログイン画面を表示しない。	
7			連携先システムB		
8			ポータル		
9		ログアウト	連携先システムA	ログアウトしたユーザは、以後システムにアクセスする際に再度認証を求められる。	
10			連携先システムB		
11			ポータル		
12		タイムアウト		一定時間画面遷移をしていない場合に、タイムアウトし、ログアウト後の状態と同じ状態になる。	
13		エラー処理		パスワードの不正や、パスワード有効期限切れ等により認証に失敗した場合、エラー画面を表示し、再度ログインを促す。	
14		連携先アクセス制御		URL 毎およびアクター毎に、連携先のアクセス許可・拒否の制御を行う	
15		HTML 書き換え		連携先システムの画面から、絶対パスを相対パスに書き換える。	工数がかかるため、対応しない
16	ID 管理	ユーザ利用	ユーザ属性変更	自分自身の属性を変更できること	
17			パスワード変更	自分自身のパスワードが変更できること	
18		管理者利用	ユーザ登録/変更/削除	1 ユーザごとの登録、変更、削除を行う。	
19			パスワードポリシ設定	パスワードを類推されにくくするため、パスワードポリシの定義を行う	

20			ユーザアカウント管理	アカウントを検索できる アカウントの停止・停止解除ができる アカウントのパスワード初期化ができる パスワードポリシの適用	
21			ID のグルーピング	ロール（1ユーザが複数持てる権限）、組織（ユーザの集合）	
22			ロールの登録/削除	ロールの登録、削除を行う。	
23			ユーザー括登録/変更/削除	CSVファイルによるユーザの一括登録、変更（サービス契約変更含む）、削除を行う。（ワークフローによる承認はなし） ※一括登録時、データチェックでエラーとなった場合、登録途中のデータも全てロールバックされる	
24		アクセス制御		ロール毎に、利用できるID管理機能を制御する	
25		多言語化		英語および日本語で利用できる。	
26		メール送信		アカウント作成、パスワード変更、パスワード期限切れのメールを送信する	
27	ポータル利用	ポータル機能利用	ダイナミックリンク	連携先システムへのリンクを提供する。	
28			お知らせ	コンテンツをユーザに表示する	
29			監査ログ参照	サイトへのアクセス履歴や、認証履歴を画面で参照できる。	
30			スケジュール	ユーザのスケジュールを管理する。	工数がかかるため、対応しない
31			ファイルライブラリ	提携フォーム等各種ファイルを添付し、ユーザのダウンロードを可能とする。	
32			掲示板	お知らせ等、ポータルの使用者に情報提供を行う。	
33			ワークフロー	ワークフロー機能を提供する。本システムでは使用しない。	
34		アクセス制御		ロール毎に、利用できるポータルの機能を制御する	
35	管理画面直接利用	OpemAM 管理コンソール利用		シングルサインオンせずに、OpenAM の管理コンソールにローカルのアカウントでログインする	
36		ポータル管理画面利用		シングルサインオンせずに、ポータルの管理画面にローカルのアカウントでログインする。	
37	連携先システム直接ログイン	連携先システムA		連携先システムAに、シングルサインオンせずに、直接ログインする	対応しない
38		連携先システム B		連携先システム B に、シングルサインオンせずに、直接ログインする	対応しない
38	プロビジョニング		ユーザ属性連携	ユーザ ID を連携先システムに連携する	工数がかかるため、対応しない

2.7. 非機能要件

2.7.1. 運用要件

項番	大分類	中分類	内容	備考

1	定常運用	データバックアップ	過失によるデータ消失などの事態を防ぐため、バックアップ/リストアできること。バックアップ対象は顧客属性情報と認証先情報とする。 毎日バックアップを取得し、1週間分バックアップを取得する。	
2		ログ出力	システムの利用状況の把握、障害やその予兆の検知、障害発生時の問題解決のため、各ミドルウェアのログを出力すること。 また、顧客が準備した監視サーバからログ監視ができるようにすること。	
3		ログ保管	ログを18カ月分保管すること。	
4		ログ参照	以下のログについて、ポータルから簡単に参照できるようにすること。 ・Web サーバアクセスログ ・認証・認可ログ	
5		稼働統計情報取得	以下の統計情報を取得し、1週間分保管する事。 ・CPU 使用率 ・メモリ使用率 ・IO 負荷状況 ・ネットワークコネクション数	
6		時刻同期	全てのサーバで時刻をそろえること	
7		サービス時間	24 時間 × 365 日サービスを継続できること。 ただし、日次運用により 10 秒間システムが使えなくなるのは許容する。	
8	監視運用	死活監視	顧客が準備した監視サーバから、以下の項目を監視できるようにすること。 ・ICMP,TCP ポート,プロセス,URL	顧客が準備した監視サーバから監視する。 本設計書では設計対象外。
9		ログ監視	顧客が準備した監視サーバから、障害を検知できるように、監視サーバから以下のログを監視できるようにすること。 ・Apache, Tomcat, OpenAM, MySQL レプリケーションログ, OpenLDAP レプリケーションログ	
10		ジョブ稼働監視	顧客が準備した監視サーバが、cron のジョブの失敗を検知できるように、ジョブの失敗をログに出力し、監視できるようにすること。	
11		リソース監視	顧客が準備したから、CPU、メモリ、ディスク使用量を監視できるようにすること。	
12	障害時運用	プロセス起動・停止	障害時に、手動でプロセスの起動・停止ができるように、手順を準備すること。	
13		MySQL・OpenLDAP データリストア	データ障害時に、バックアップデータからデータを復旧できるように、手順を準備する事。	
14	特別時運用	ソフトウェアバージョンアップ	各ソフトウェアのバージョンアップ手順を準備する事	今回は対象外
15		リリース手順	コンテンツのリリース等、定型的なリリースの手順を準備する事	今回は対象外

2.7.2. セキュリティ要件

項番	大分類	中分類	内容	備考
1	監査		以下の情報について、過去 18 カ月分さかのぼり、参照可能とする。 参照できるのはリアルタイムではなく、前日分以前とする。 ・更新操作履歴 ・組織への配属や、ロール付与などに関する履歴 ・シングルサインオンの認証情報 ・シングルサインオンのアクセス制御情報 ・各システムへの Web アクセスログ ・ユーザ属性情報	
2	暗号化	通信暗号化	機密情報は暗号化して通信すること。	
3		データ暗号化	バックアップデータは本番環境内にある場合は暗号化しなくてよい。	
4		パスワード暗号化	パスワード暗号化して格納すること。	
4	不正アクセス防止		必要以上のユーザにアクセスをさせないようにすること。	
5	ウイルス対策		OS 上のファイルがウイルスに感染していないか定期的に検査する。	セキュリティホールのリスクの低い Linux サーバで構成されているため、ウィルス対策は行わない。
6	ガイドライン準拠		総務省の「ASP・SaaS における情報セキュリティ対策ガイドライン」に準拠すること	工数がかかるため、対応しない

2.7.3. 性能要件

項番	大分類	中分類	小分類	内容	備考
1	O/L 処理性能	シングルサインオン	ログイン	【ピーク時】(※) ・リクエストからログイン完了まで 3 秒以内に処理が終わる事 ・処理件数は 10 件/秒以上処理できること ・5000 ユーザが既にログインしている状態で上記の性能を満たす事 【通常時】 ・リクエストからログイン完了まで 3 秒以内に処理が終わる事 ・処理件数は 1 件/秒以上処理できること ・500 ユーザが既にログインしている状態で上記の性能を満たす事	
2			連携先システムアクセス(プロキシ)	【ピーク時】(※) ・連携先のからの応答を受け取り、画面描画が始まるまでの時間が 3 秒以内にである事 ・処理件数は 50 件/秒以上処理できること ・5000 ユーザが既にログインしている状態で上記の性能を満たす事 【通常時】 ・連携先のからの応答を受け取り、画面描画が始まるまでの時間が 3 秒以内にである事 ・処理件数は 5 件/秒以上処理できること ・500 ユーザが既にログインしている状態で上記の性能を満たす事	
5		ID 管理		要件無し	プロジェクト毎に検討する事
7		ポータル利用		要件無し	
8		管理画面アクセス		要件無し	
9	B/T 処理性能	ユーザー括登録		要件無し	プロジェクトごとに検討する事。 記載する場合は、〇〇件を〇〇時間で登録できることという書き方にすること
10	キャパシティ	登録ユーザ数		10000 ユーザ登録されている状態で、上記性能を満たせること。	
11		連携先数		要件無し	プロジェクト毎に検討する事

※ピーク時は1日の中で5分間存在する最も負荷の高い時間とする。それ以外の時間は通常時とする。

2.7.4. 耐障害性要件

項番	大分類	中分類	内容	備考

1	冗長化		ハードウェアやソフトウェア障害に陥った際でも、システムが停止しないこと	障害が発生した場合でもその影響が連携先システム全体の機能不全にはつながらないため、冗長化は行わない
2	耐ハードウェア障害		HW 交換後に復旧手順に従い、システム復旧できること	仮想マシンであるため、H/W 故障は考慮しない。
3	耐ソフトウェア障害	OS	復旧手順に従い作業を行うことにより、OS を正常に起動できること	
4		プロセス	復旧手順に従い作業を行うことにより、プロセスを正常に起動できること	
5	耐データ障害	バックアップリストア	復旧手順に従い作業を行うことにより、、データ復旧できること。 また、データ復旧した時に、全営業日の状態に戻れること。	
6	許容する障害対応時間		24 時間以内にシステム復旧できること	

2.7.5. 拡張要件

項番	大分類	中分類	内容	備考
1	連携先追加		手順に従い、連携先追加できること。	
2	データ量拡張		ディスク拡張によりデータ量を拡張できること。	
3	処理性能拡張		リソースの割り当てを増やすなど、スケールアップにより拡張できること。	

2.7.6. 移行要件

項番	大分類	中分類	内容	備考
1	移行対象範囲		（センター切り替え or サーバ移行 or プロダクト移行）	データ移行は発生しないため検討不要
2	移行実施日（期間）		（リハーサル日程、リリース日程）	
3	移行方針		（段階移行か一括移行か） （並行稼働の有無）	

2.7.7. 端末要件

OS ＼ ブラウザ		IE7	IE8	IE9	Chrome	Opera	FireFox ESR	Safari
					（最新版が対象）			
Windows	XP(SP3 以上)							
	Vista(SP2 以上)	○	○				○	
	7		○	○			○	
	8							
MacOS	Mac OS X 10.6							
	Mac OS X10.7							

	Mac OS X 10.8							
iOS	iOS5							
Android	4.0							

2.7.8. 維持管理・サポート要件

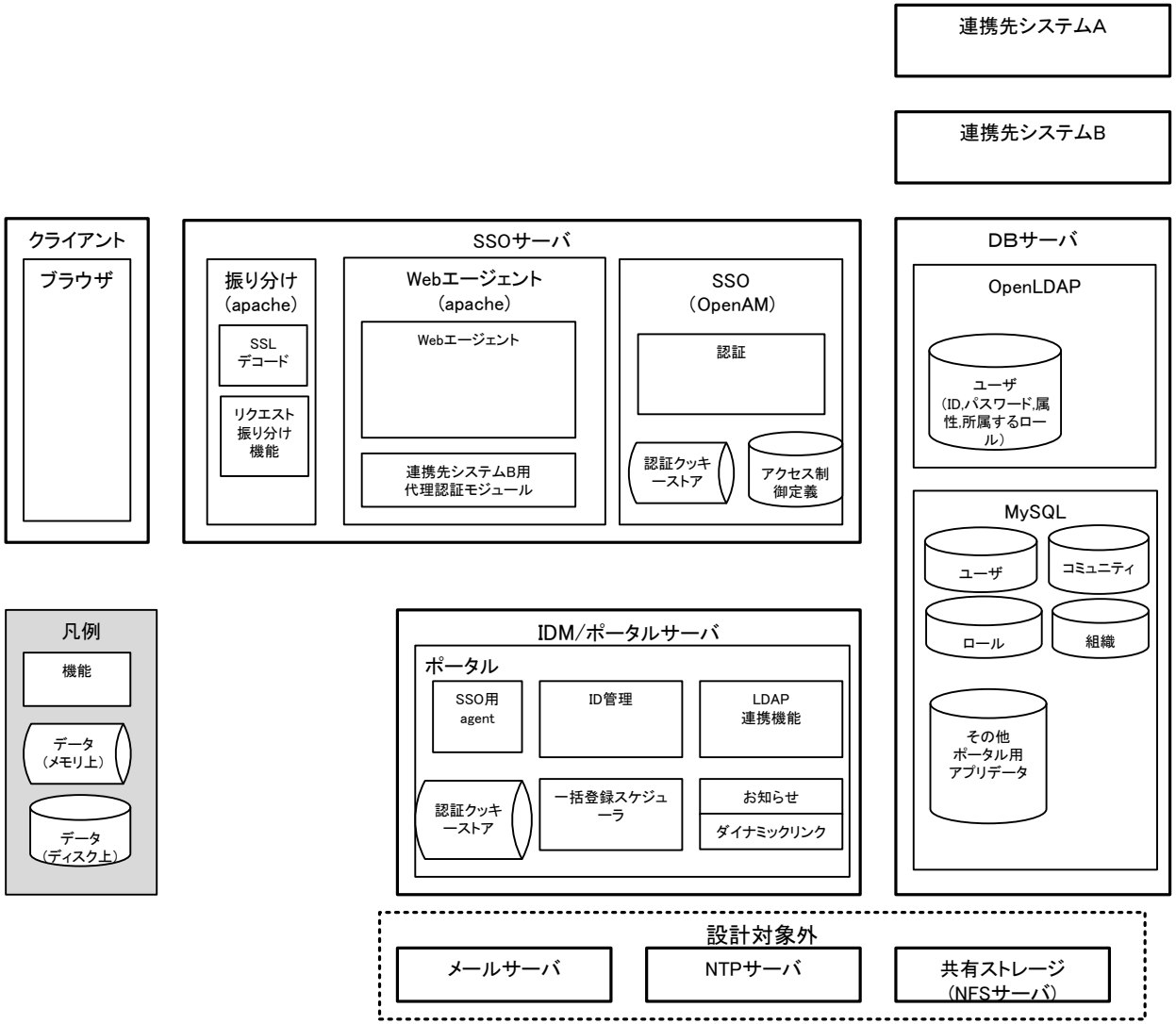
項番	大分類	中分類	内容	備考
1	維持管理・サポート		リリース後の維持管理(トラブルシューティング)は顧客主体で行えるように準備する。	

2.7.9. 開発要件

項番	大分類	中分類	内容	備考
1	開発		本番と同じ機能を有するステージング環境を作成し、そこで方式設計書のテストを実施する事。ただし、性能などの非機能要件はステージング環境で満たさなくてよい。 方式設計以下のドキュメントの開発・テストについては、ベンダの手持ちの開発環境で実施しても構わない。	

3. システム構成図

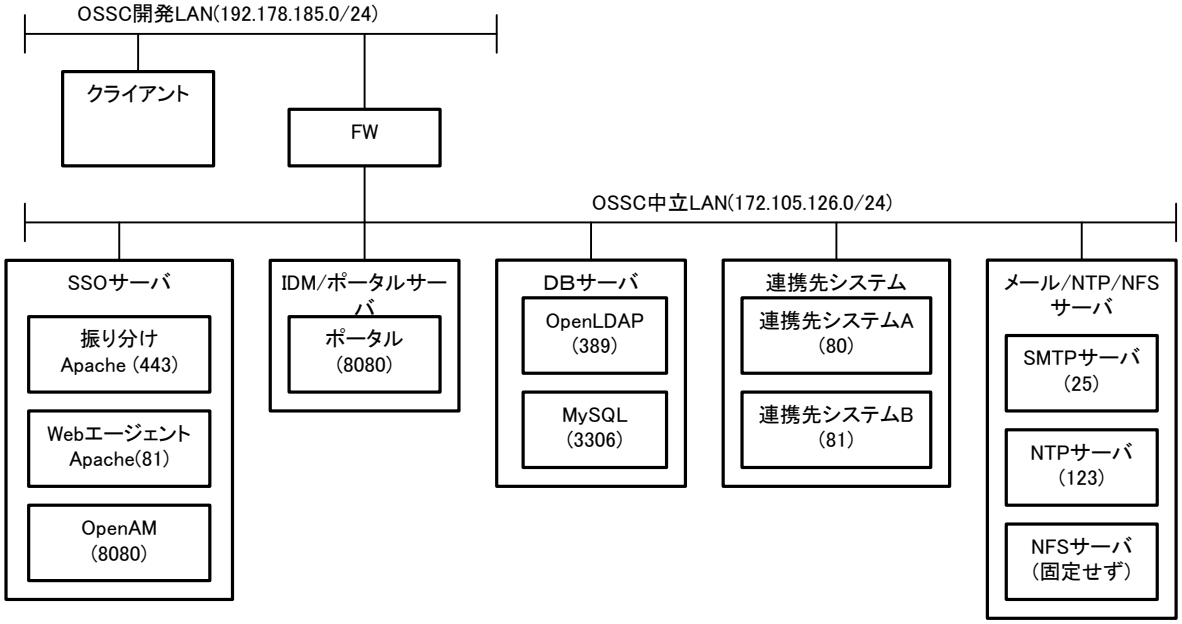
3.1. 全体構成図



3.2. ネットワーク構成

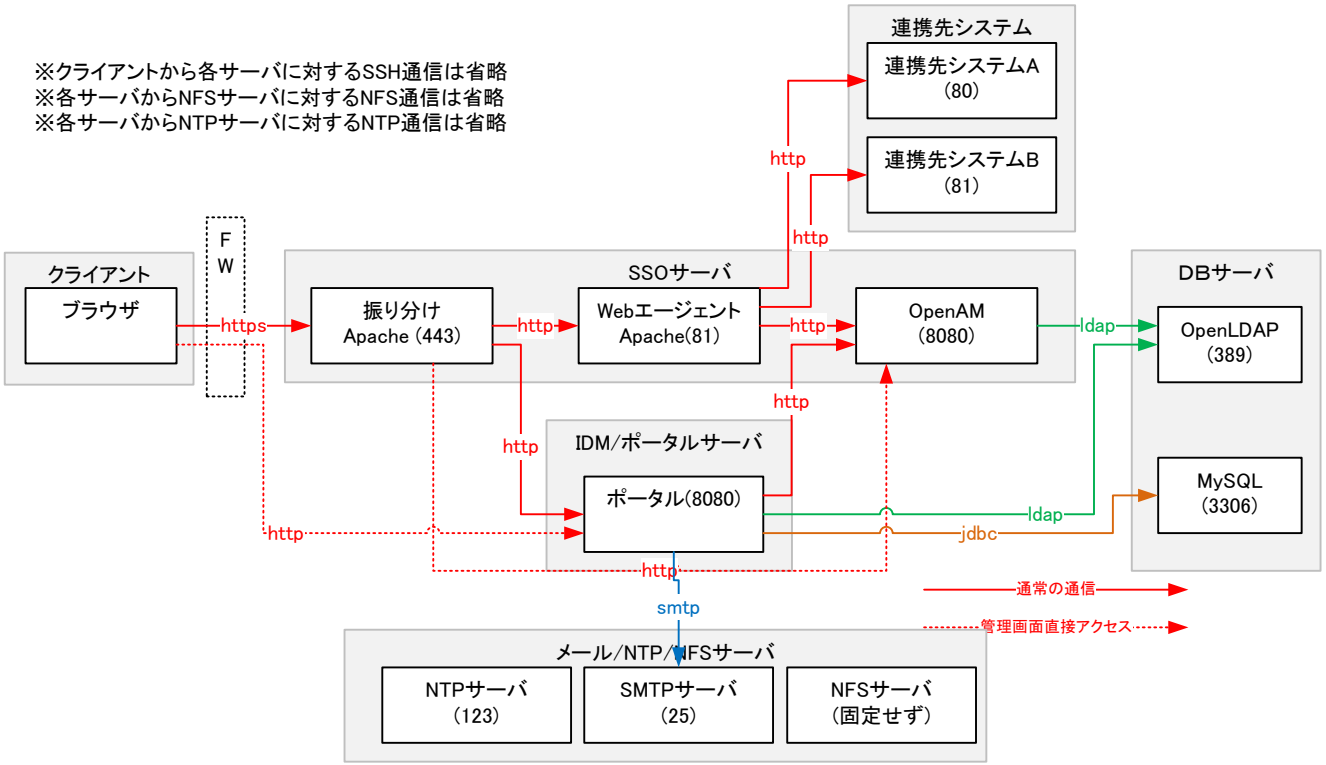
3.2.1. ネットワーク構成図

※括弧内はポート番号



3.2.2. 通信経路一覧

※クライアントから各サーバに対するSSH通信は省略
※各サーバからNFSサーバに対するNFS通信は省略
※各サーバからNTPサーバに対するNTP通信は省略



項番	プロトコル	通信元		方向	通信先		説明
		説明	ポート		説明	ポート	
1	HTTPS	ブラウザ	any	→	振り分け Apache	443	
2	HTTP	振り分け Apache	any	→	Web エージェント Apache	81	リクエスト振り分け
3	HTTP	Web エージェント Apache	any	→	OpenAM	8080	認可
4	ldap	OpenAM	any	→	OpenLDAP	389	ユーザデータ参照
5	HTTP	Web エージェント Apache	any	→	連携先システム A	80	
6	HTTP	Web エージェント Apache	any	→	連携先システム B	81	
7	HTTP	振り分け Apache	any	→	ポータル	8080	リクエスト振り分け

項番	プロトコル	通信元		方向	通信先		説明
		説明	ポート		説明	ポート	
8	HTTP	ポータル	any	→	OpenAM	8080	認可
9	ldap	ポータル	any	→	OpenLDAP	389	ユーザの認証データ CRUD
10	jdbc	ポータル	any	→	MySQL	3306	ポータルのデータ CRUD
11	HTTP	ブラウザ	any	→	ポータル	8180	管理コンソール直接利用
12	HTTP	振り分け Apache	any	→	OpenAM	8080	管理コンソール直接利用
13	SMTP	ポータル	any	→	SMTP サーバ	25	メール送信
14	NTP	SSO サーバ	any	→	NTP サーバ	123	時刻同期
15	NTP	IDM/ポータルサーバ	any	→	NTP サーバ	123	時刻同期
16	NTP	DB サーバ	any	→	NTP サーバ	123	時刻同期
17	SSH	ブラウザ	any	→	SSO サーバ	22	サーバ管理用
18	SSH	ブラウザ	any	→	IDM/ポータルサーバ	22	サーバ管理用
19	SSH	ブラウザ	any	→	DB サーバ	22	サーバ管理用
20	NFS	SSO サーバ	any	→	NFS サーバ	未固定	NSF マウント
21	NFS	IDM/ポータルサーバ	any	→	NFS サーバ	未固定	NSF マウント
22	NFS	DB サーバ	any	→	NFS サーバ	未固定	NSF マウント

3.3. ハードウェア構成

項番	サーバ	HW 型番	CPU	メモリ	ディスク容量	IP アドレス	備考
1	SSO サーバ	仮想マシン	Intel Corei7 2core	4G	20G	172.105.126.101	
2	IDM/ポータルサーバ	仮想マシン	Intel Corei7 2core	4G	20G	172.105.126.102	
3	DB サーバ	仮想マシン	Intel Corei7 2core	4G	20G	172.105.126.103	
4	連携先システム	仮想マシン	Intel Corei7 2core	4G	20G	172.105.126.100	
5	メール/NTP サーバ					172.105.126.110	設計対象外

3.4. ソフトウェア構成

3.4.1. ソフトウェア選定の考え方

メジャーバージョン、マイナーバージョンについては、実績があるものを採用する。
リビジョンはセキュリティの観点から最新のものを採用する。

3.4.2. ソフトウェアー覧

SSOサーバ

			代理認証モジュール	OpenAM 9	運用スクリプト
mod_ssl	mod_proxy	エージェント	mod_perl	Tomcat 6	rbatch
apache 2.2(振り分け)		apache 2.2(Webエージェント)	perl	JDK 1.7	ruby 1.9.3
CentOS 6.3					

IDM/ポータルサーバ

OpenStandia Portal	運用スクリプト
JBossAS 4.2.3	rbatch
JDK 1.6	ruby 1.9.3
CentOS 6.3	

DBサーバ

OpenLDAP 2.4	MySQL5.5	運用スクリプト
		rbatch
		ruby 1.9.3
CentOS 6.3		

項番	サーバ	ソフトウェア	バージョン	備考
1	SSO サーバ	Red Hat Enterprise Linux	6.3	
2		JDK	1.7	
3		Apache httpd	2.2.24	振り分け用。OSSC 独自でソースからコンパイル
4		Apache httpd	2.2.24	エージェント用。OSSC 独自でソースからコンパイル
5		Tomcat	6.0.35	
6		OpenAM	9.5.5	
7		Web Policy エージェント	3.0.4	
8		perl	5.10.1	OS 付属のバージョン
9		mod_perl	–	代理認証モジュールに付属
10		mod_ssl	–	Apache httpd に付属
11		ruby	1.9.3	運用スクリプト実行用
12		rbatch	–	Ruby ベースの運用スクリプトフレームワーク
13	IDM/ポータルサーバ	Red Hat Enterprise Linux	6.3	
14		JDK	1.7	
15		JBoss AS	4.2.3	
16		OpenStandia Portal	5.2.3	
17		ruby	1.9.3	運用スクリプト実行用
18		rbatch	–	Ruby ベースの運用スクリプトフレームワーク
19	DB サーバ	Red Hat Enterprise Linux	6.3	

20		OpenLDAP	2.4.25-42.el6	
21		MySQL	5.5.30	
22		ruby	1.9.3	運用スクリプト実行用
23		rbatch	-	Ruby ベースの運用スクリプトフレームワーク

4. 処理方式

機能一覧いついて、それぞれ処理方式を記載する

4.1. シングルサインオン

4.1.1. 対応方針

4.1.2. ログイン

（１） 概要

SSO には、システム構成により以下の3つの構成をとることができる。

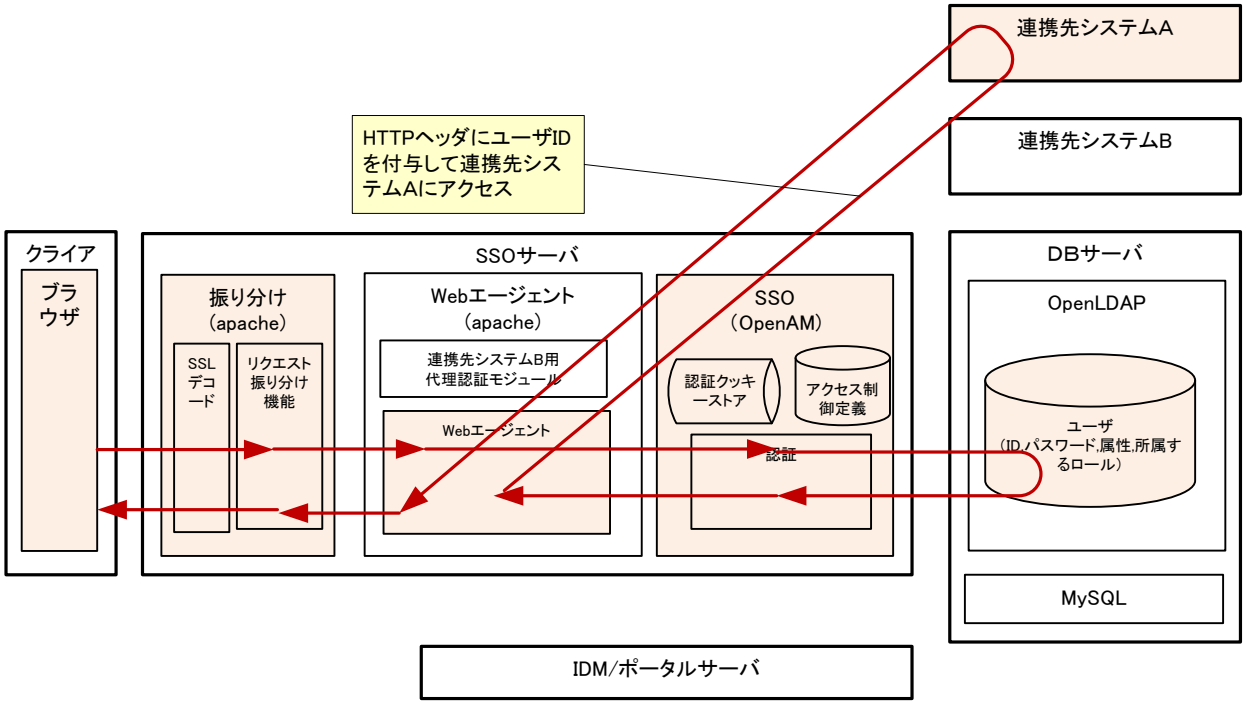
- リバースプロキシ型 ・・・ 共用リバースプロキシサーバから、各サービスへ転送しシングルサインオンを実現する。
- エージェント型 ・・・ 各サービスにポリシーエージェントを配置しシングルサインオンを実現する。
- SAML 型 ・・・SAML を用いてシングルサインオンを実現する。

今回は振り分けサーバを用いるため、リバースプロキシ型のみを採用する。

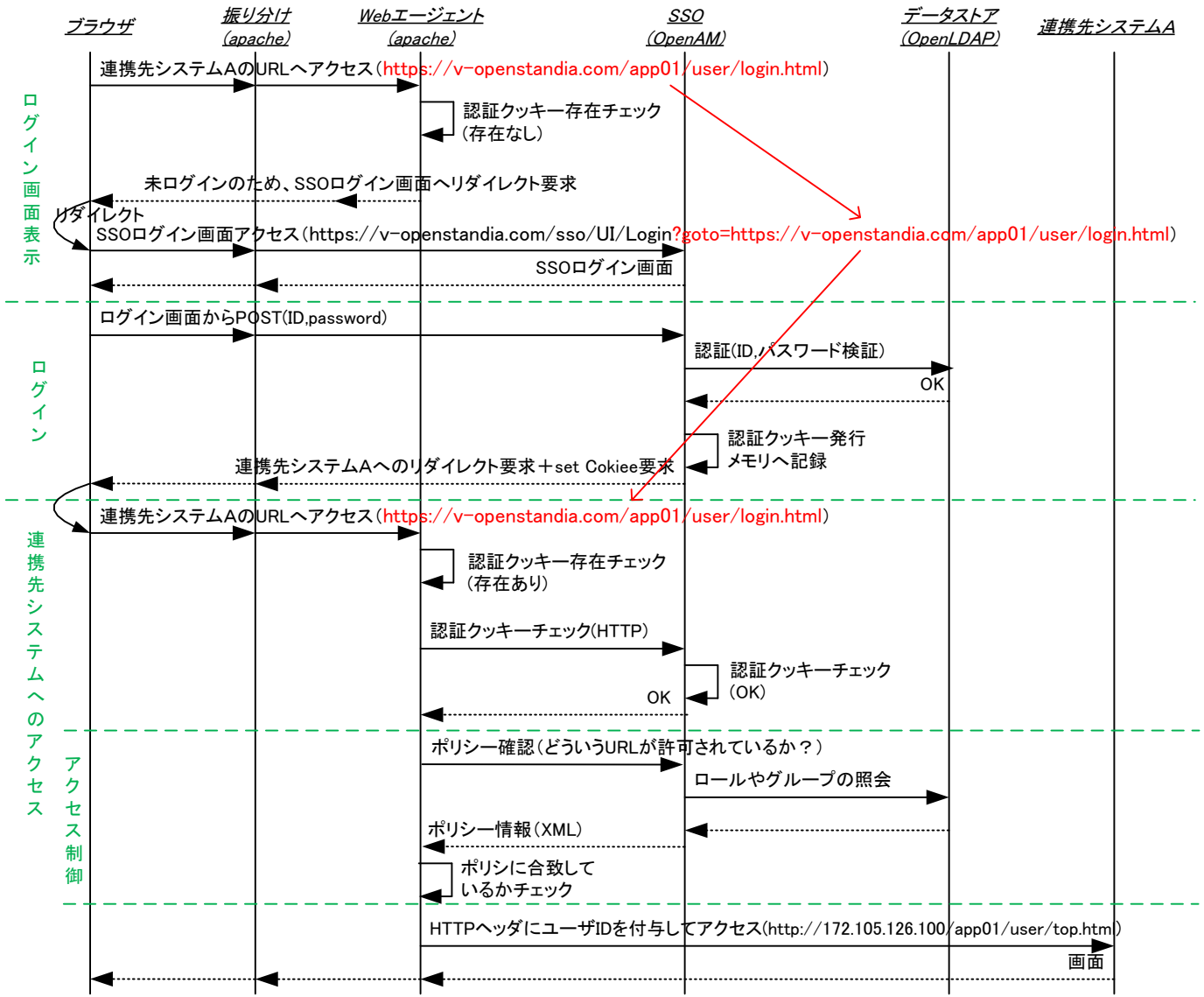
（２） 連携先システムＡログイン（リバプロ方式）

処理概要

SSO サーバが連携先システム A にリクエストにアクセスする際、HTTP ヘッダにユーザ ID を入れることにより、1回のリクエストでログイン可能状態にする。



処理シーケンス

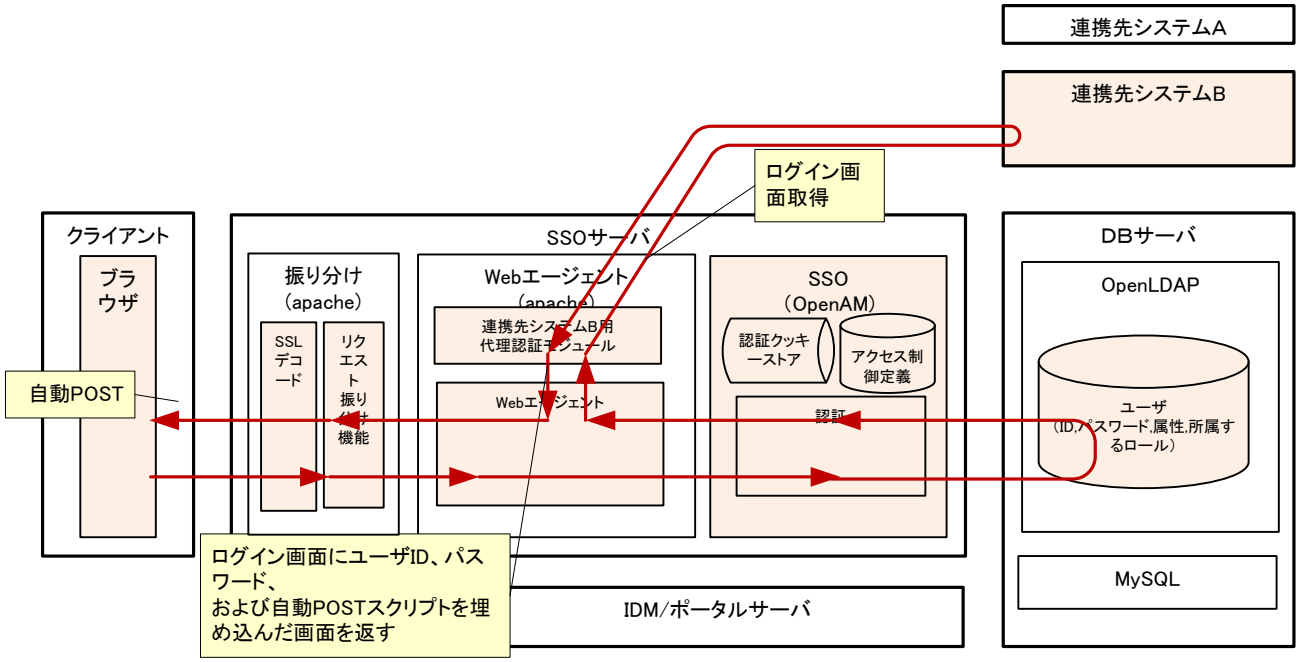


連携先システムへのパラメータ

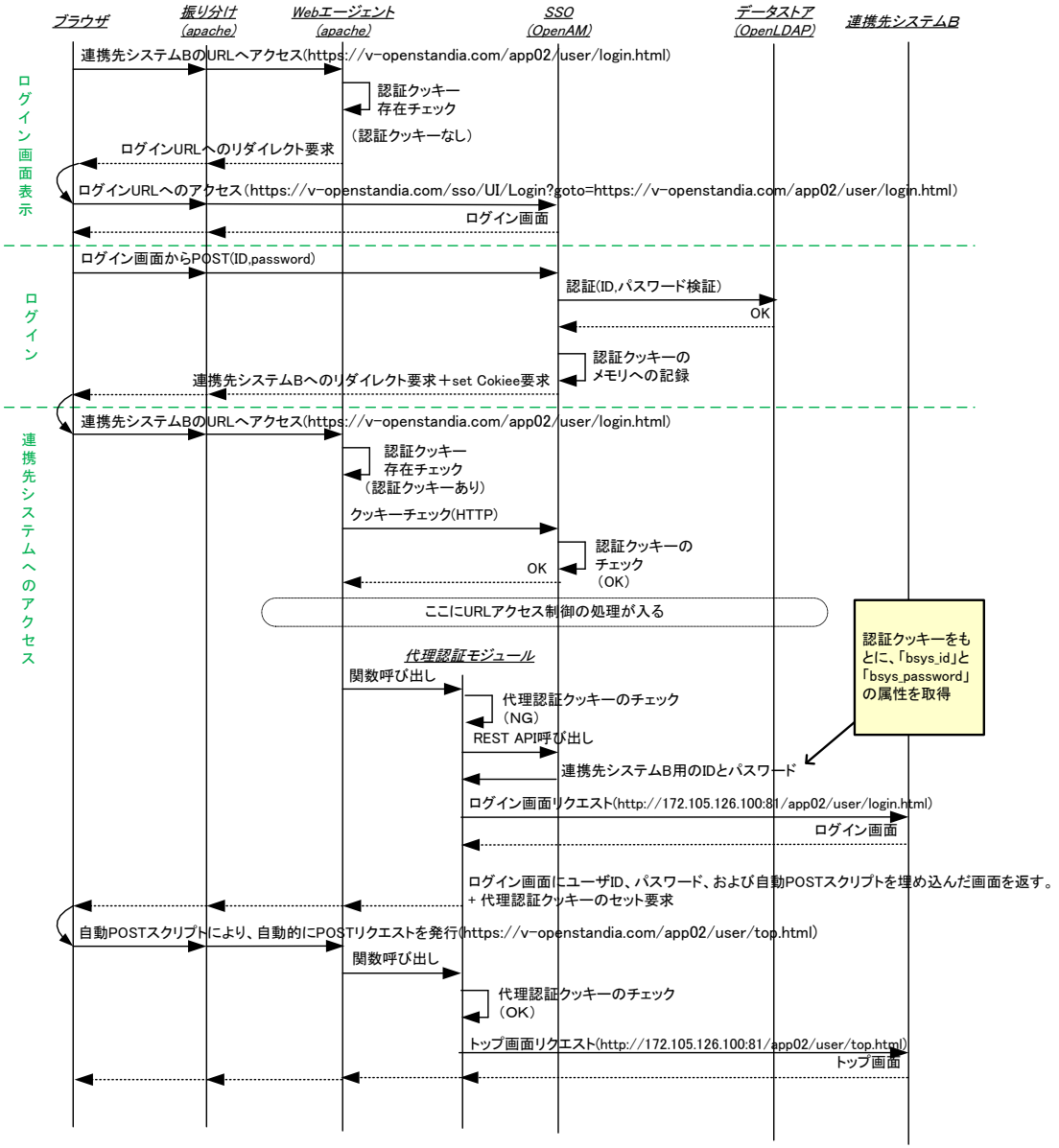
連携先システム	画面	URL	HTTP ヘッダキー	HTTP ヘッダバリュー	説明
Aシステム	トップ画面	http://172.105.126.100/app01/user/top.html	USER_ID	A システムのユーザ ID	

（３） 連携先システムＢログイン（リバプロ+代理認証方式）

処理概要



処理シーケンス

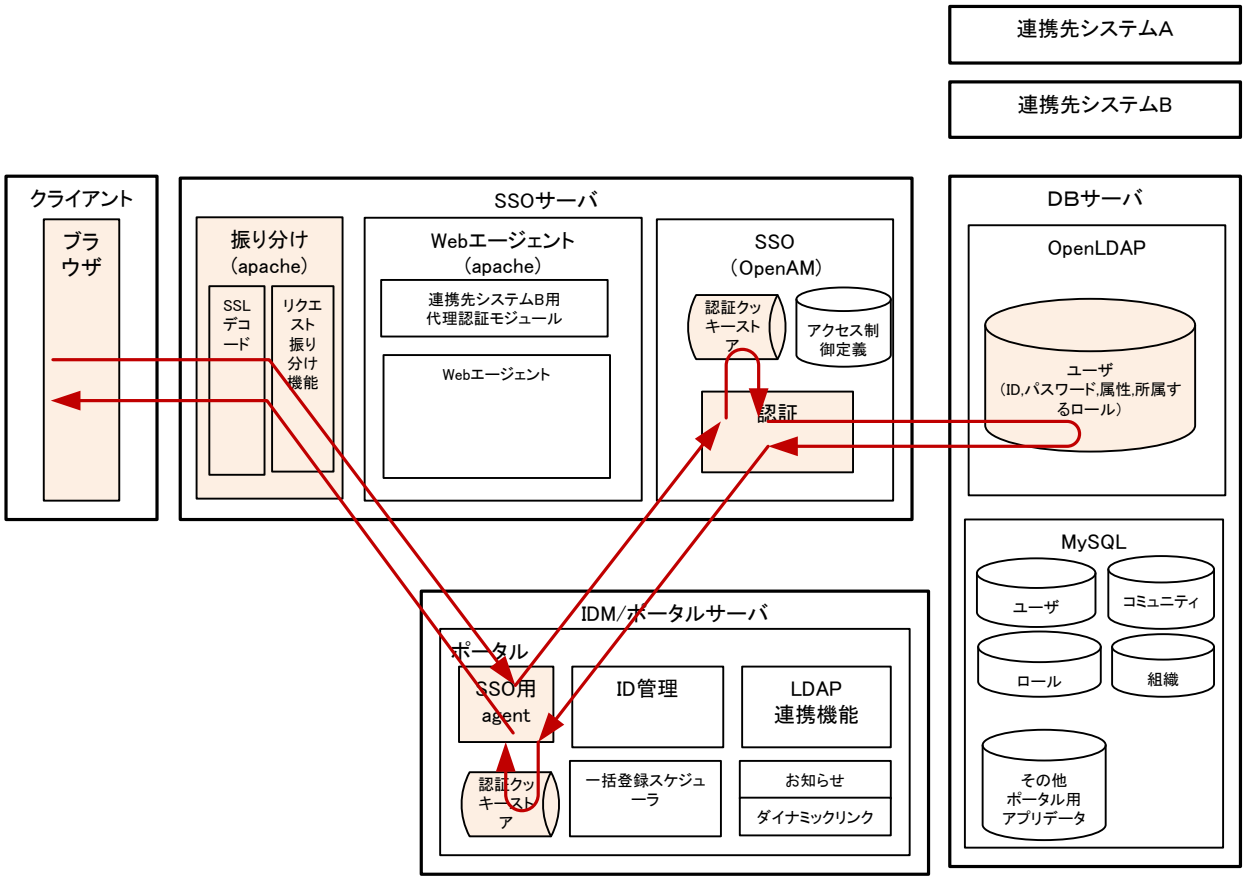


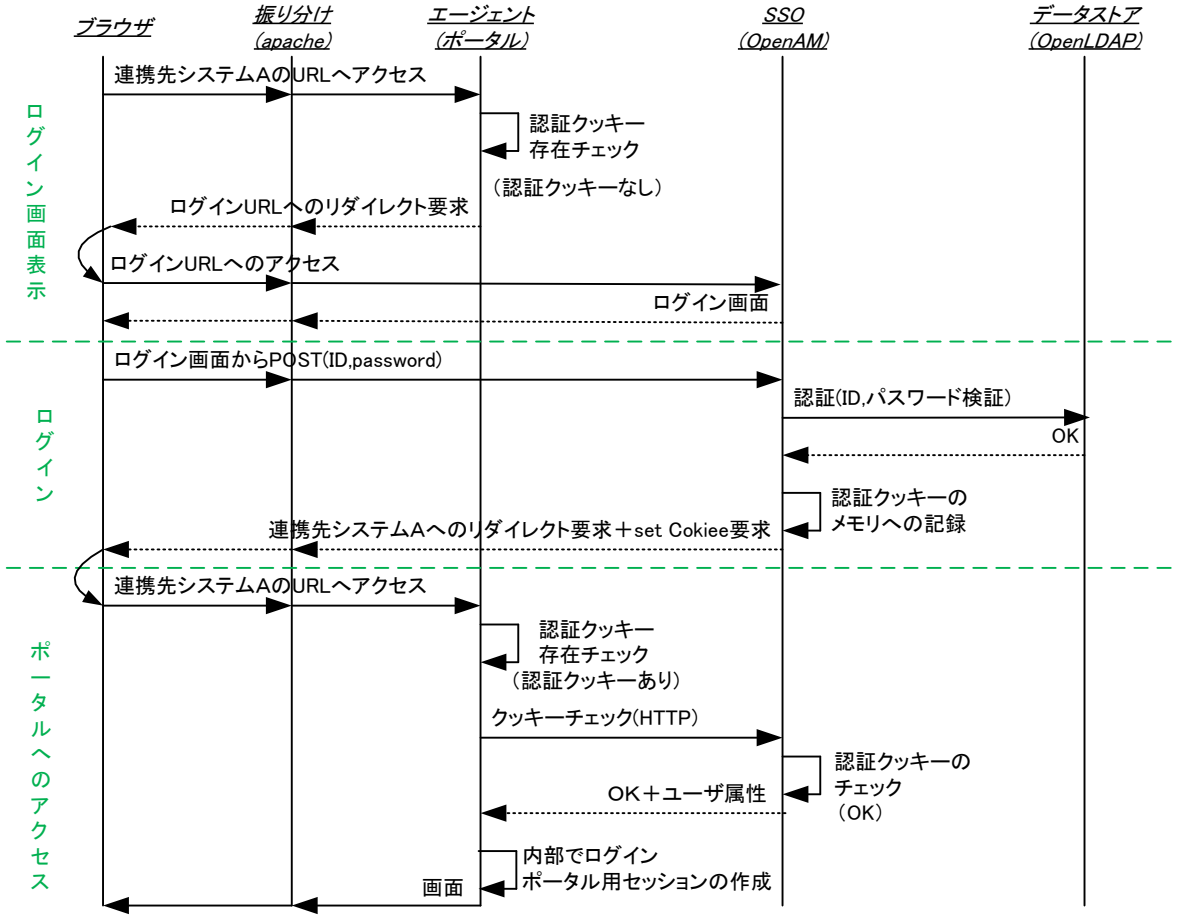
連携先システムへのパラメータ

連携先システム	画面	URL	補足情報
Bシステム	ログイン画面	http://172.105.126.100:81/app02/user/login.html	・ログインフォーム名は「form01」 ・この URL であれば、代理認証クッキーの有無にかかわらず、代理認証処理を行う（強制ログイン URL 設定）
	トップ画面（POST 先）	http://172.105.126.100:81/app02/user/top.html	・POST パラメータ ID=(連携先システム B 用 ID) PASSWORD=(連携先システム B 用パスワード) ・LDAP 上の対応する属性 ID : bsys_id PASSWORD : bsys_password

※詳細は別紙「認証インターフェース」参照

（４） ポータルログイン（エージェント型）



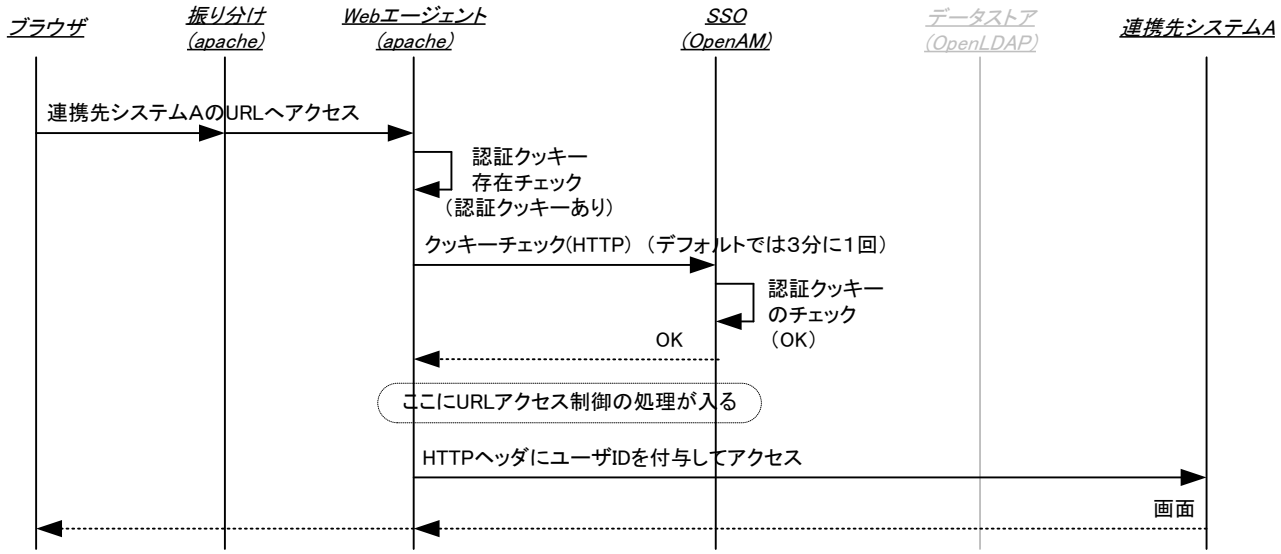


※リバプロ方式との違い

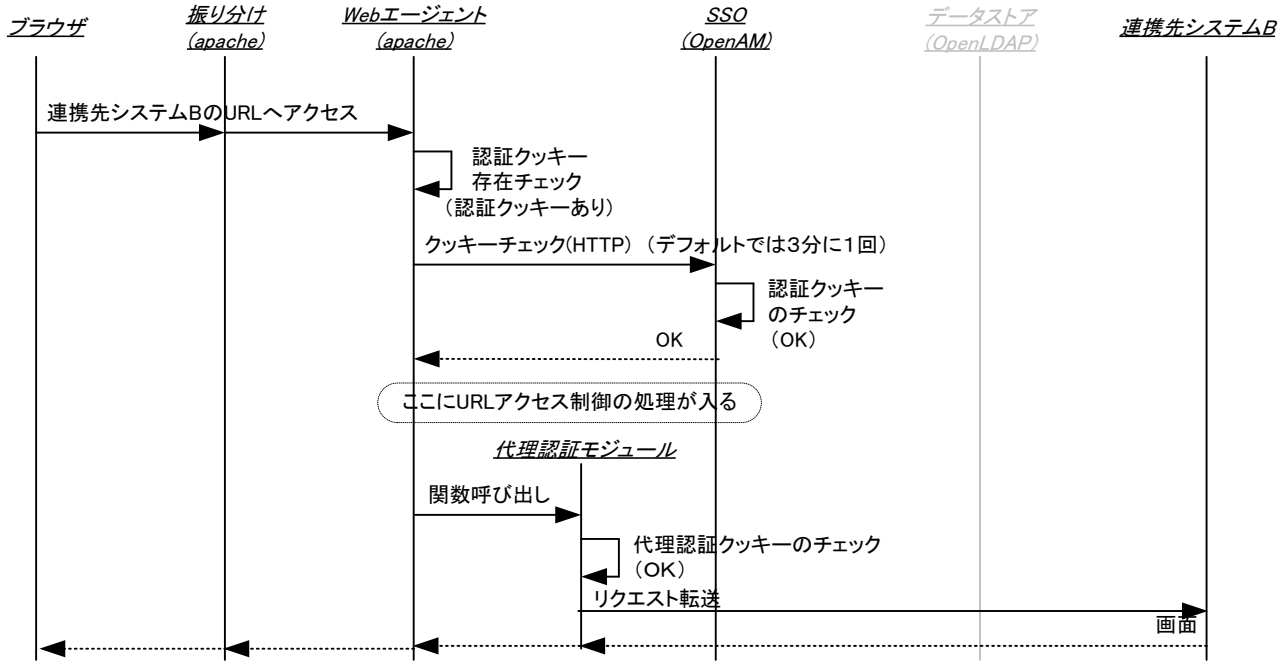
- ポータル自身がエージェントを持っていること
- アクセス制御がないこと

4.1.3. ログイン中チェック（クッキーチェック）

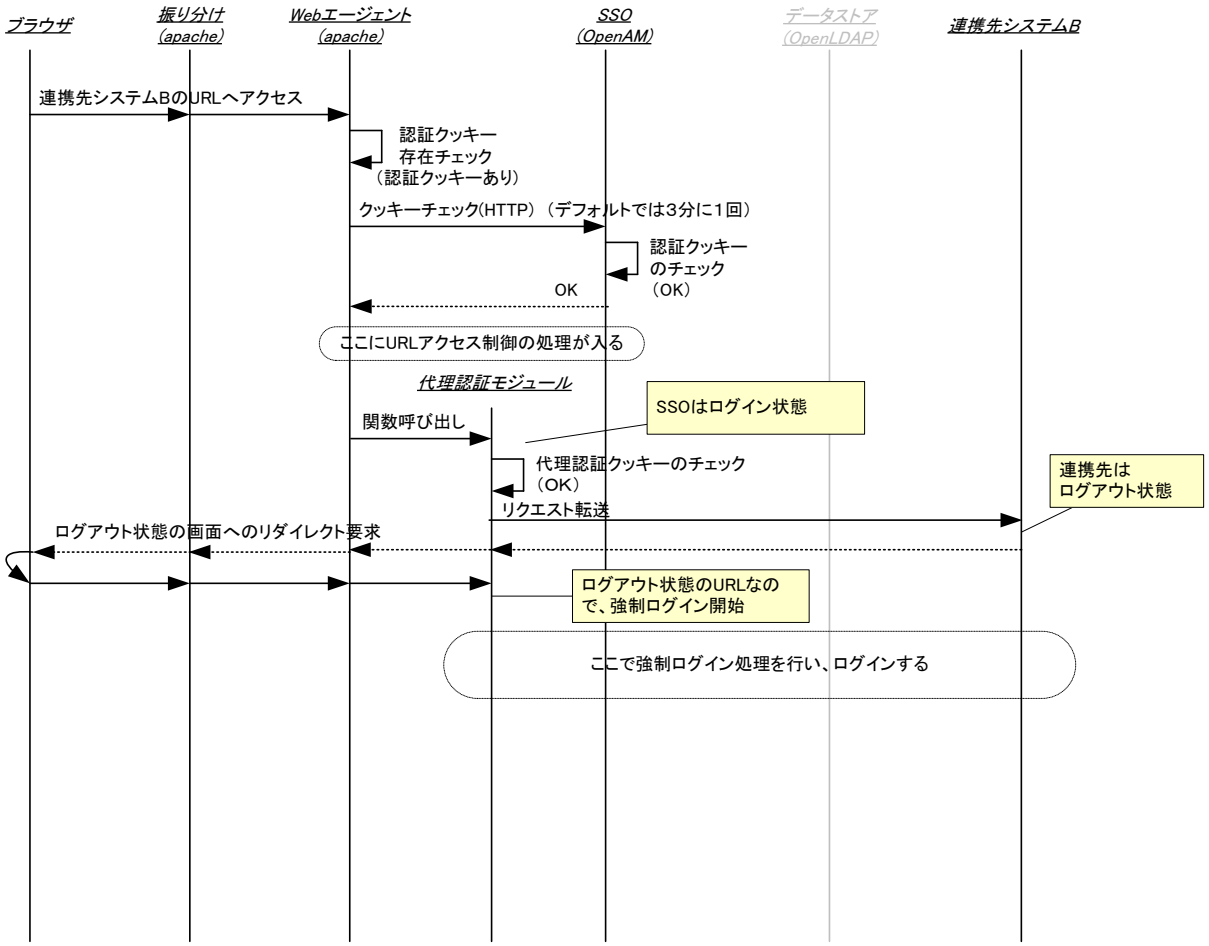
（１） 連携先システムＡ（リバプロ方式）



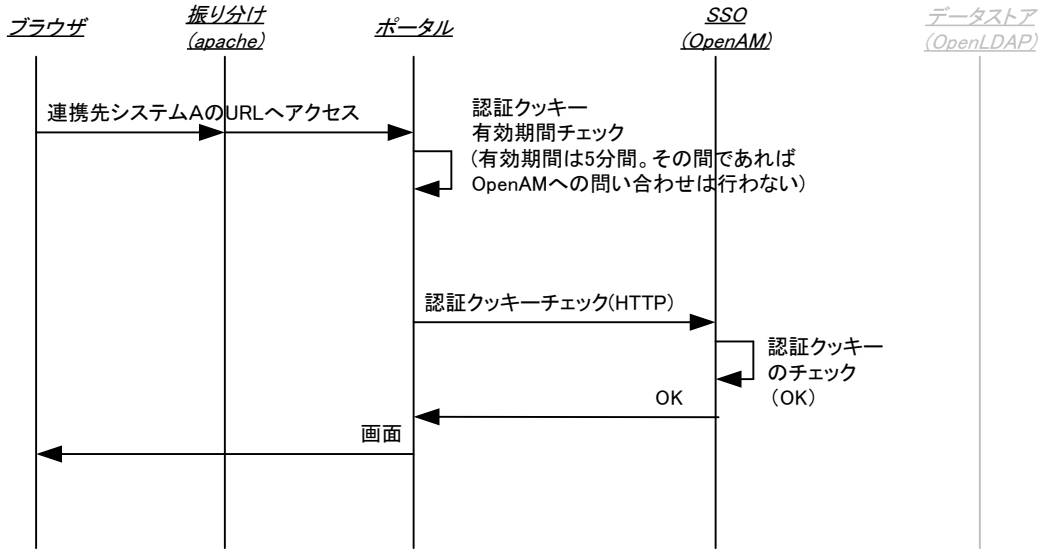
（２） 連携先システムＢ（リバプロ＋代理認証方式）



- 強制ログイン
  - SSO がログイン状態であっても、連携先システムがログアウト状態になった場合、ユーザに連携先システムのログイン画面が表示されてしまう問題がある。
  - この問題を解決するために、連携先システムがログアウト状態である時のURLをあらかじめ調査し、そのURLへリダイレクトされるような応答があった場合には、SSO がログイン済みであっても再度ログイン処理を行う。これを強制ログイン処理と呼ぶ。
  - 強制ログイン処理のシーケンスは以下の通り



（３） ポータル



- 強制ログイン
  - SSO がログイン状態中にポータルがログアウト状態（セッションタイムアウト、サーバダウンなどで）となった場合でも、ポータルの機能で強制ログインを行う。
  - ただし、ポータルのセッションは破棄されているため、作業中の状態はクリアされる。

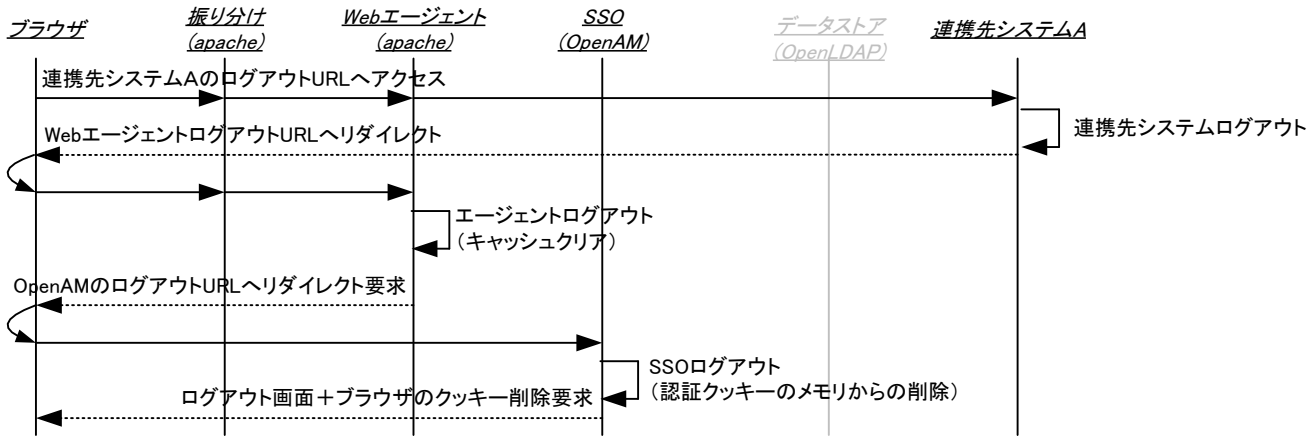
4.1.4. ログアウト

（１） 連携先システムＡ（リバプロ方式）

ログアウト処理とは、認証クッキー／セッションを削除し、ログイン状態を解除する処理である。
SSO ログアウト完了画面には連携サービスやブランドによって異なるページに遷移する為の URL リンクを表示する。

ログアウト処理は連携サービスログアウト／エージェントログアウト／SSO ログアウトの3つが存在する。

No	ログアウト処理	説明
1	連携先システムログアウト	連携先システムにてログアウトを行う。
2	エージェントログアウト	Web エージェントの SSO セッションのキャッシュをクリアするする。
3	SSO ログアウト	SSO サーバ（OpenAM）の認証クッキーを削除する。 Web ブラウザに対して認証クッキー削除指示（set-cookie）を行う。

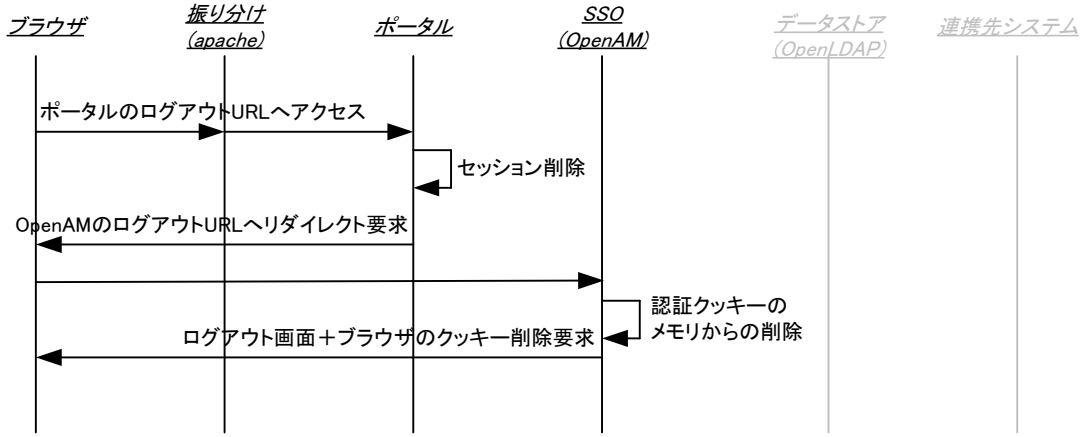


（２） 連携先システムＢ（リバプロ＋代理認証方式）

連携先システムAと同じ

（３） ポータル

ポータルのログアウトは Web エージェントを経由しないので、以下のようなフロー になる



4.1.5. タイムアウト

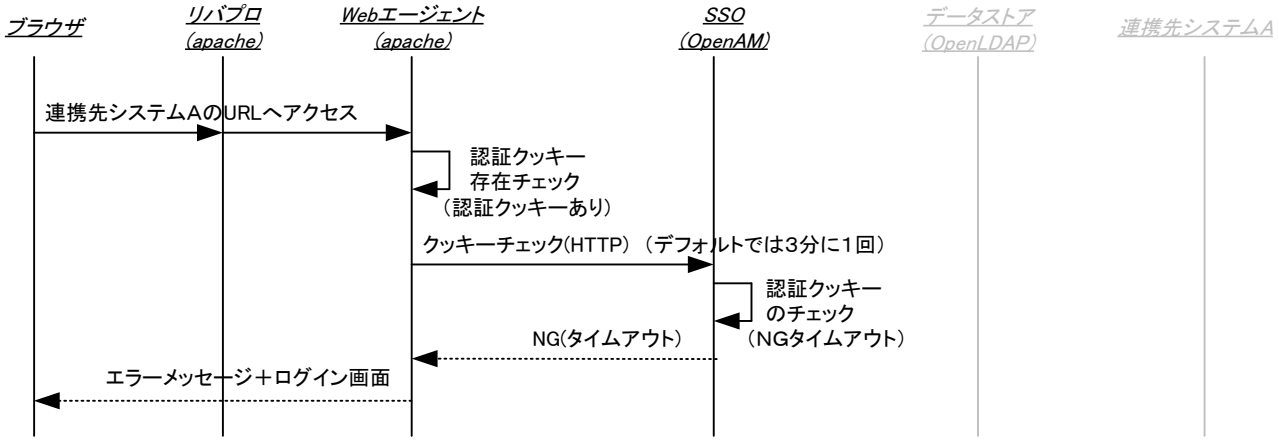
各システムのタイムアウトの時間は以下の関係性を満たす必要がある。ただし、連携先が自動ログイン機能を備えている、かつ、セッションリセットによる影響をユーザが許容可能な場合はその限りではない。

最大アイドル時間 ＜ 最大セッション時間 ＜連携先システムのタイムアウト値の最小値

そのため、以下の通りのタイムアウト時間とする。

サーバ	タイムアウトの種類	設定値	説明	備考
SSO サーバ	最大アイドル時間	60 分	一定時間以上無操作状態が続いた場合	タイムアウトした場合、再リクエスト時に SSO ログイン画面へ遷移する
	最大セッション時間	120 分	操作有無に関わらず操作時間が一定時間（ログイン時刻から一定時間経過）以上に達した場合	タイムアウトした場合、再リクエスト時に SSO ログイン画面へ遷移する
連携先システムA	セッションタイムアウト	なし		
連携先システムB	セッションタイムアウト	180 分	ユーザが 180 分何も操作を行わない場合	
ポータル	セッションタイムアウト	121 分	一定時間以上無操作状態が続いた場合	SSO サーバの最大セッション時間より長く設定する

SSO サーバタイムアウトの場合のシーケンス



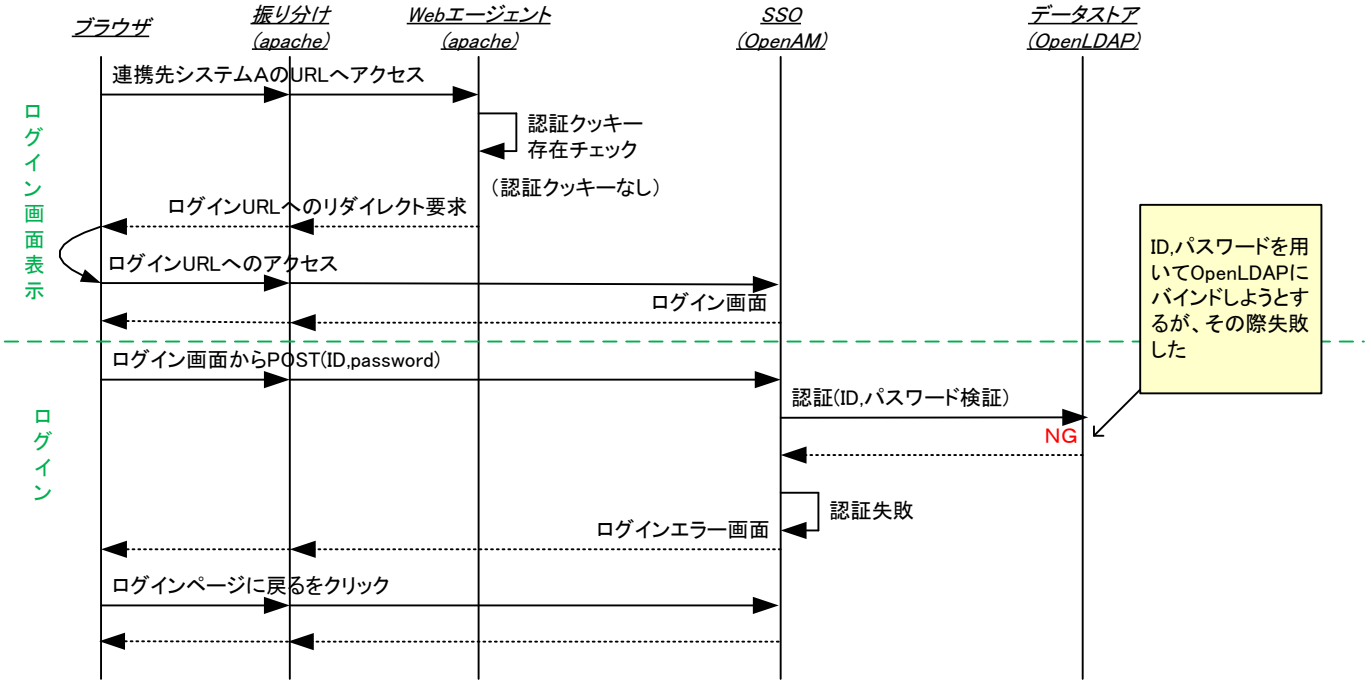
4.1.6. エラー処理

エラー処理は認証エラー、URL アクセス拒否、不正 URL の3つがある

（1） 認証エラー

認証エラーは、ログイン時にクライアントから送られてきた ID とパスワードを用いて、OpenLDAP で認証（OpenLDAP にバインド）を行い、それが失敗した場合のエラーである。

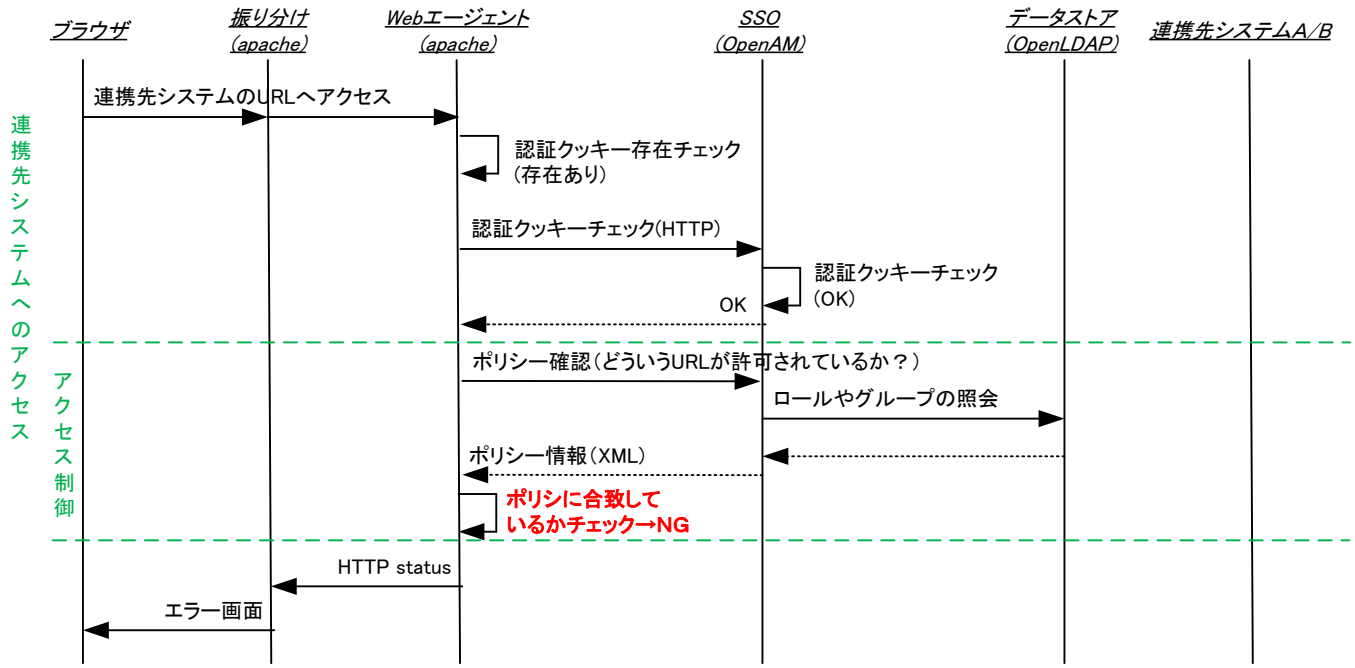
シーケンス



（２） URL アクセス拒否

URL アクセス拒否は、OpenAM の URL アクセス制御機能にて拒否された場合のエラーである。

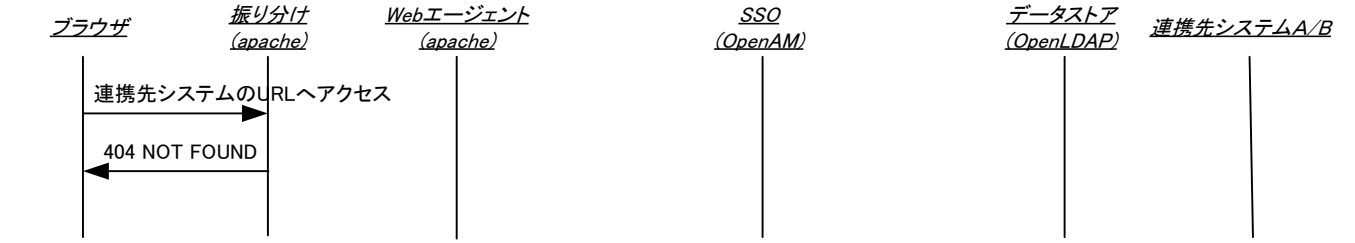
シーケンス



（３） 不正ＵＲＬ

不正 URL は、振り分け(apache)にて設定されていない不正な URL にアクセスした場合のエラーである。

シーケンス



#	カテゴリ	想定されるエラーケース	エラー画面出力箇所	本システムでの動作	ユーザ対応	備考
1.	認証エラー	ログイン認証時に、存在しないログイン ID でログイン	OpenAM	ログインエラー画面表示	登録されている ID でログインする	
2.		ログイン認証時に、パスワード入力誤り	OpenAM	ログインエラー画面表示	正しいパスワードでログインする	
3.		ログイン認証時に、パスワード入力を指定回数誤り。アカウントロックアウトされた。	OpenAM	ログインエラー画面表示	アカウントロックアウトが解除されるまで 30 分待つ	本システムではアカウントロック機能なし
4.		アカウントロック中のユーザがログイン	OpenAM	ログインエラー画面表示	アカウントロックアウトが解除されるまで 30 分待つ	本システムではアカウントロック機能なし
5.		アカウント停止中のユーザがログイン	OpenAM	ログインエラー画面表示		
6.	アクセス拒否	未ログイン状態で認証済のみ許可するページにアクセス	Web エージェント(Apache)	ログイン画面へ遷移する	ログインする。	
7.		ログイン済状態で許可されていないページにアクセス。	Web エージェント(Apache)	振り分け(Apache)にて 403 FORBIDDEN 用のエラー画面を表示	無し	
8.	不正 URL	存在しないページにアクセス	振り分け(Apache)	振り分け(Apache)にて 404 NOT FOUND 用のエエラー画面を表示	存在するページにアクセスする	

4.1.7. URL アクセス制御

OpenAM による URL レベルのアクセス制御機能を用いて、アクター毎、URL 毎にアクセス制御を行う。
詳細は別紙参照「URL 別アクセスコントロール」参照。
また、URL アクセス制御の説明については付録15.3アクセス制御の説明を参照

URL ＼ アクター		A システムユーザ	A システム管理者	Bシステムユーザ	Bシステム管理者	ID 管理者	システム管理者
A システム	ユーザ画面	○	○	○	○	○	○
	管理画面	×	○	×	×	×	○
B システム	ユーザ画面	○	○	○	○	○	○
	管理画面	×	×	×	○	×	○
ポータル	ダイナミックリンク・お知らせ画面	○	○	○	○	○	○
	管理画面(※)	○	○	○	○	○	○

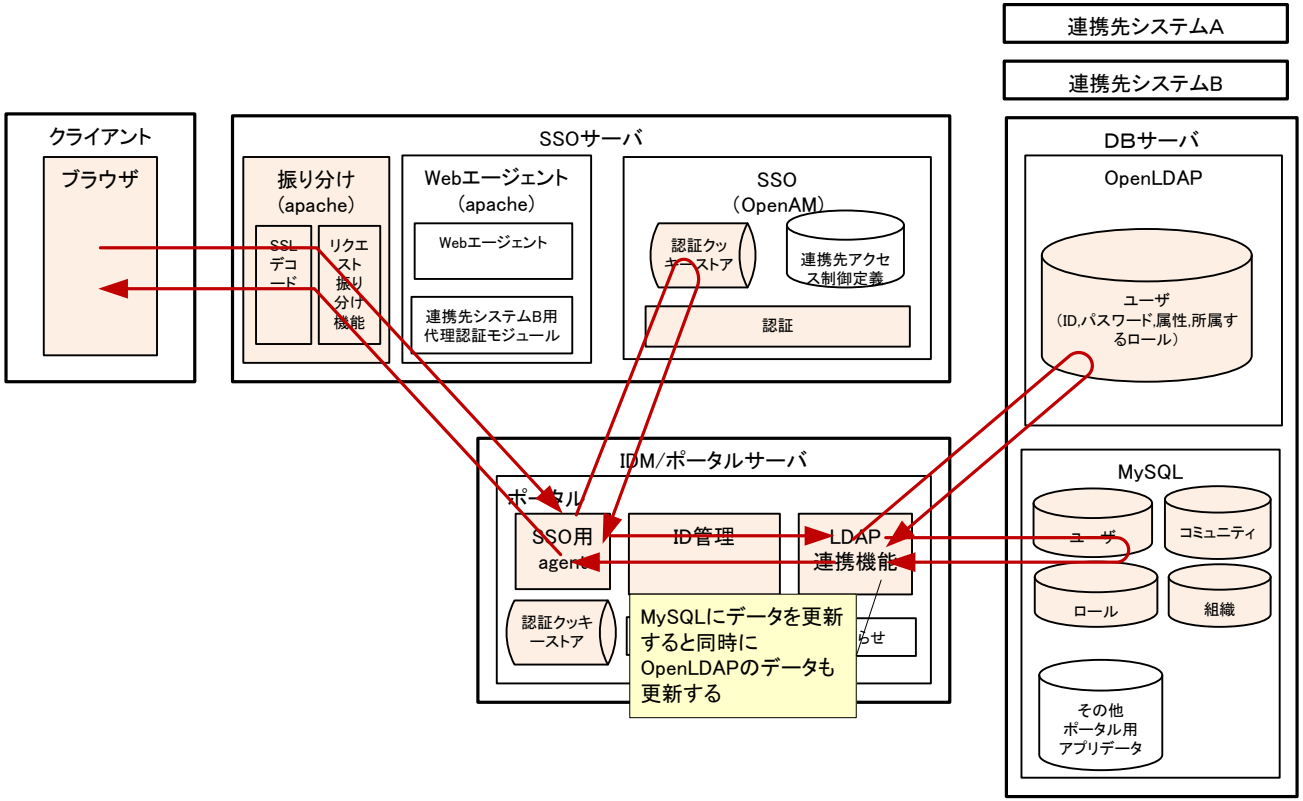
※)ポータルの管理画面 URL へは全てのアクターでもアクセスでき、セキュリティ上問題があるように思われるが、ポータルのアクセス制御機能でアクターごとにアクセス制御をかけるため、問題ない。

4.2. ID 管理

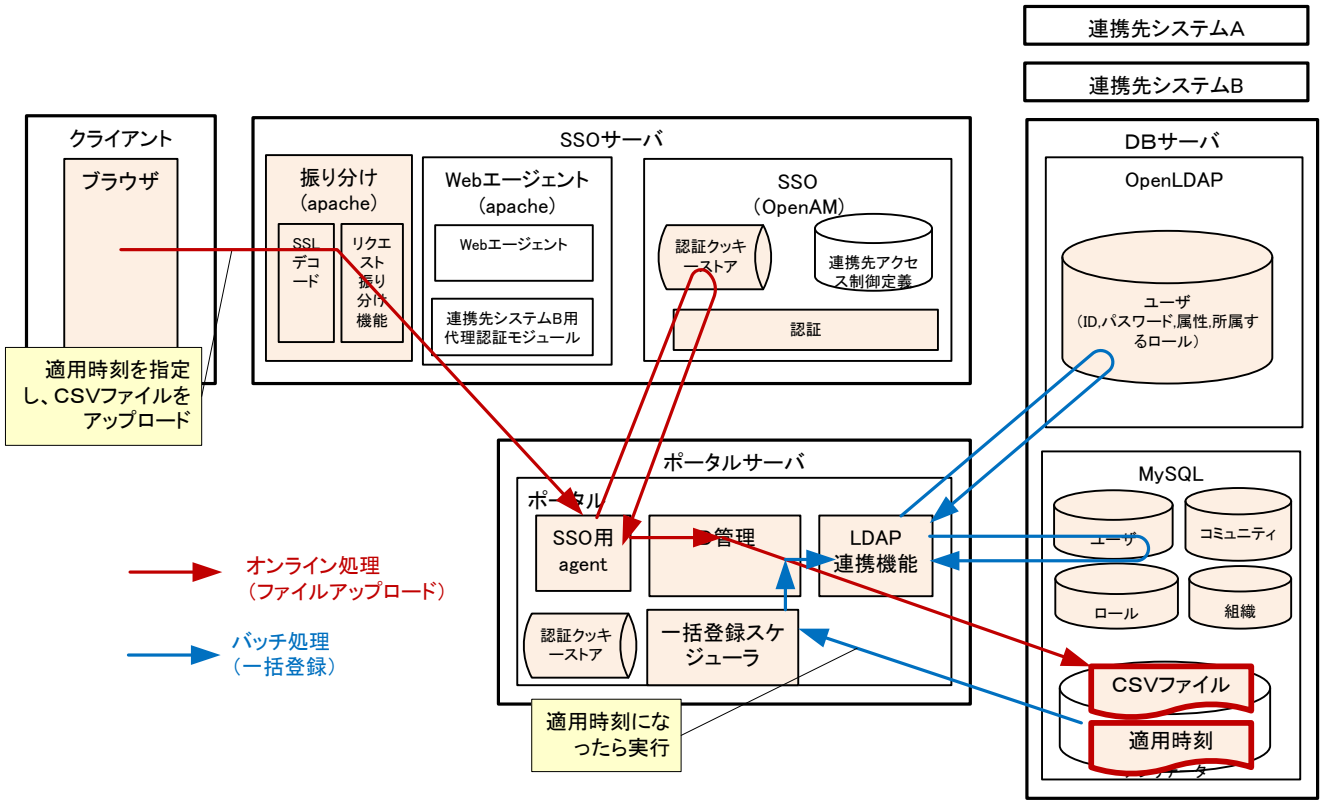
4.2.1. ID 管理処理シーケンス

オンラインの ID 管理には以下の機能があるが、いずれも同じ処理シーケンスである。

（１） ユーザ属性変更、パスワード変更、ユーザ登録/変更/削除、ユーザアカウント管理、ID のグルーピング、ロールの登録/削除、パスワードポリシ設定

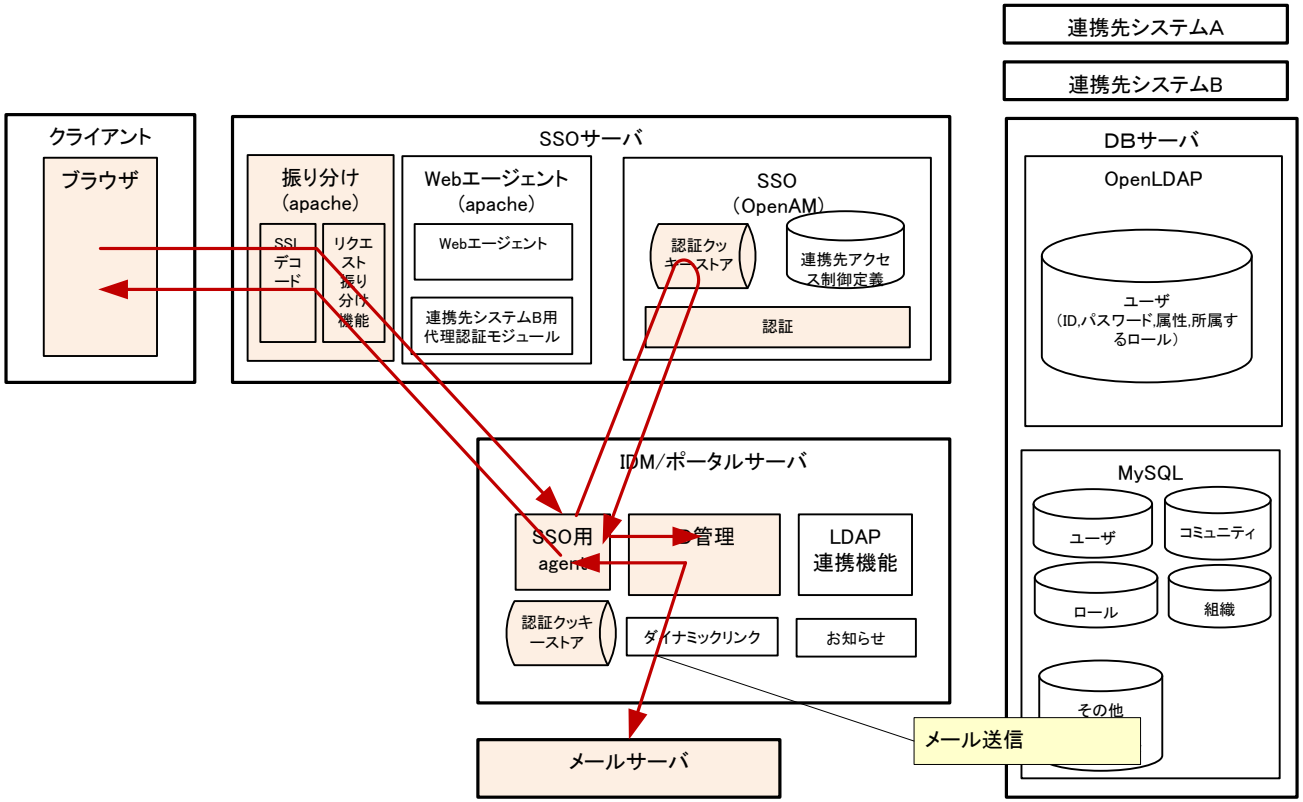


（２） ユーザー括登録/変更/削除



（３） メール送信

ポータルからパスワード初期化メールを送付する。送付先は既存のメールサーバであり、メールを受信する仕組みについては設計対象外である。



4.2.2. パスワードポリシ設定

以下のようにパスワードポリシを設ける

項番	ポリシ名	値	備考
1	パスワードの有効期限	3 か月	
2	パスワードの有効期限切れメール送付タイミング	1 日前 1 週間前	
3	パスワード文字列・長さ	・英数必須、6 文字以上(大文字小文字の区別は無し) ・スクリーンネームと同じものは NG ・メールアドレスと 6 文字部分一致するものは NG	

4	アカウントロック	10 分間でパスワードを 5 回間違った場合アカウントがロックする。30 分で復旧する。	

4.2.3. アクセス制御

ポータルのアクセス制御機能により、画面毎、アクター毎にアクセス制御をする。

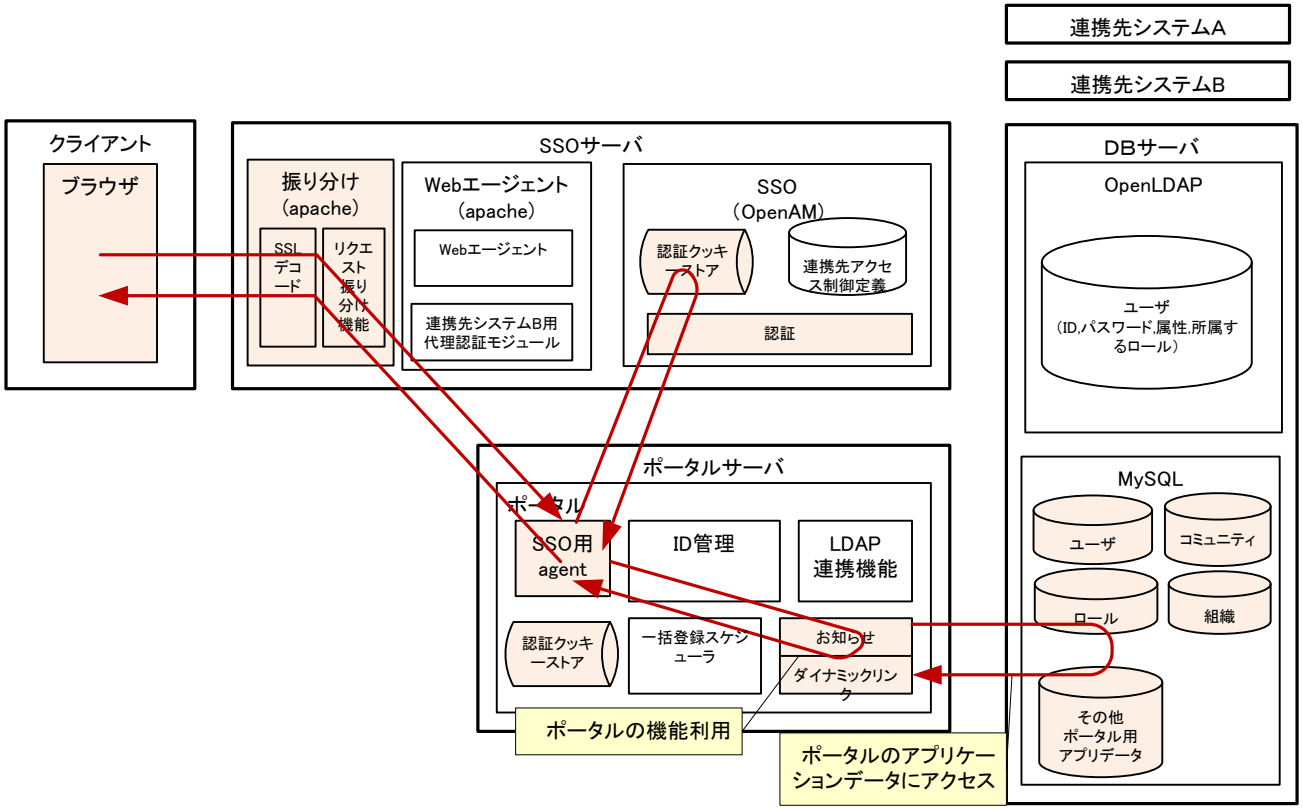
画面＼アクター		A システムユーザ	A システム管理者	Bシステムユーザ	Bシステム管理者	ID 管理者	システム管理者
ポータル	管理画面	×	×	×	×	○	○
	それ以外の画面	○	○	○	○	○	○

アクターとユーザ・コミュニティ・ロールの対応については、別紙「アクターとユーザ・コミュニティ・ロールの対応」参照。

4.2.4. 多言語化

英語と日本語に対応する。

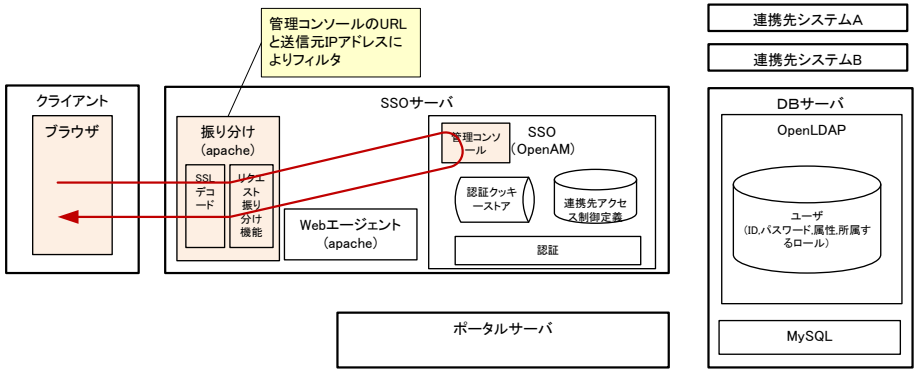
4.3. ポータル機能利用



4.4. 管理画面直接利用

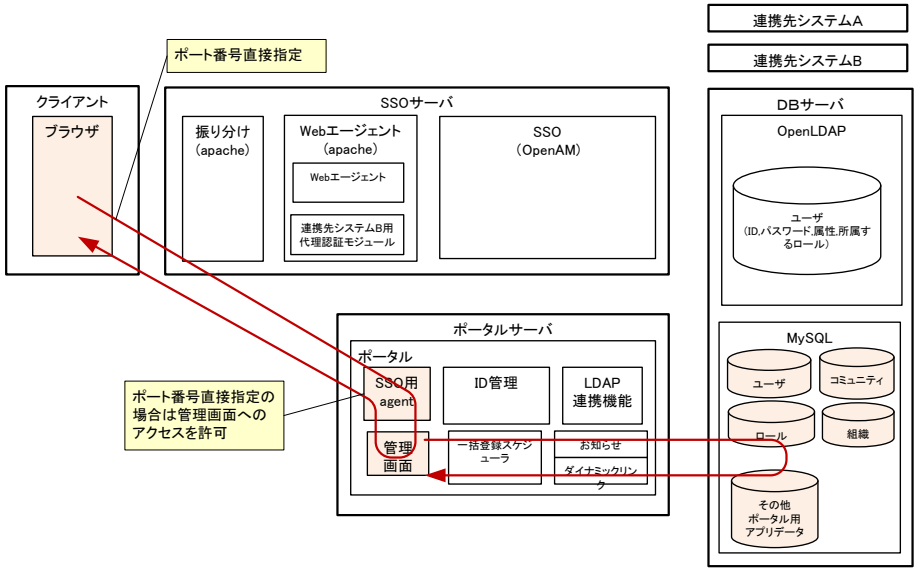
4.4.1. OpemAM 管理コンソール利用

シングルサインオンをせずに、直接 OpenAM の管理コンソールにログインする。ログインユーザのアカウント情報は、OpenAM がもつローカルのアカウント情報を用いる。



4.4.2. ポータル管理画面利用

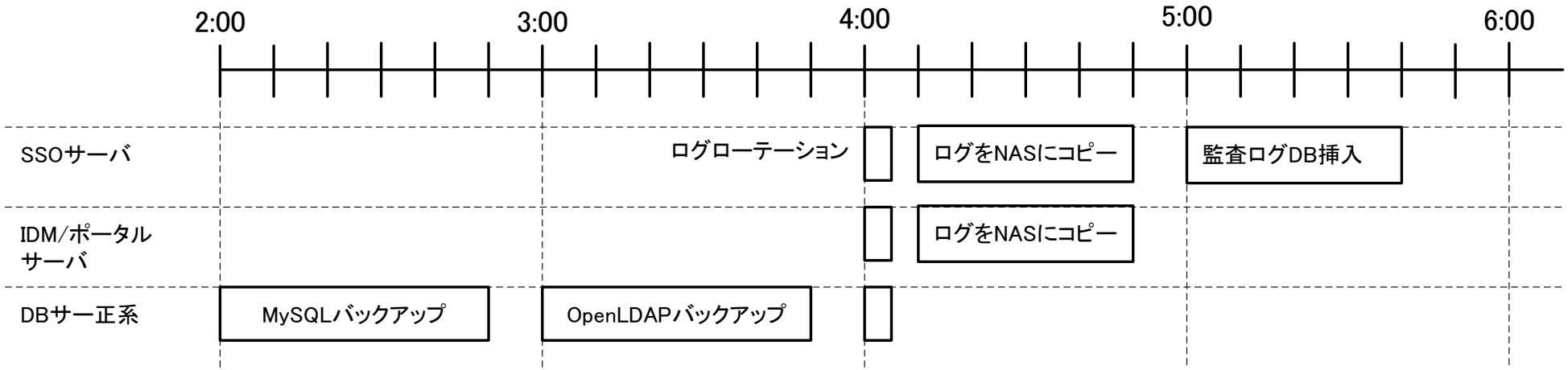
シングルサインオンをせずに、直の管理コンソールにログインする。ログインユーザのアカウント情報は、OpenAM がもつローカルのアカウント情報を用いる。



5. 運用設計

5.1. 定常運用

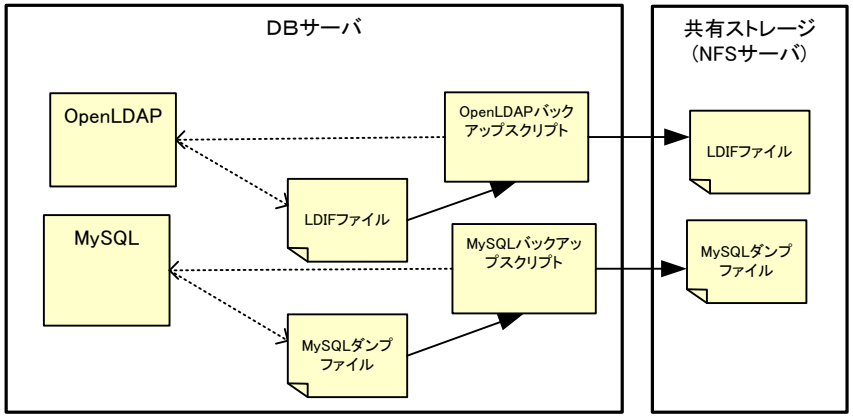
タイムチャートは以下の通り



5.1.1. データバックアップ

以下の通りバックアップを行う。

サーバ	バックアップ対象プロダクト	バックアップ対象	バックアップ先	バックアップ取得方法	タイミング	保管期間	備考
DB サーバ	OpenLDAP	OpenLDAP データディレクトリ	共有ストレージのバックアップディレクトリ	slapcat コマンドによりテキストファイルにデータを書き出し、共有ストレージにコピーする	日次 午前 2 時	7 日分	注意）更新がかかっている最中に、バックアップが行われると、論理的に不整合なバックアップになる可能性がある。OpenLDAP の slapcat コマンドにトランザクションの概念がないため。
	MySQL	全テーブル	共有ストレージのバックアップディレクトリ	mysqldump コマンドにてテキストファイルにデータを書きだし、共有ストレージにコピーする	日次 午前 3 時	7 日分	

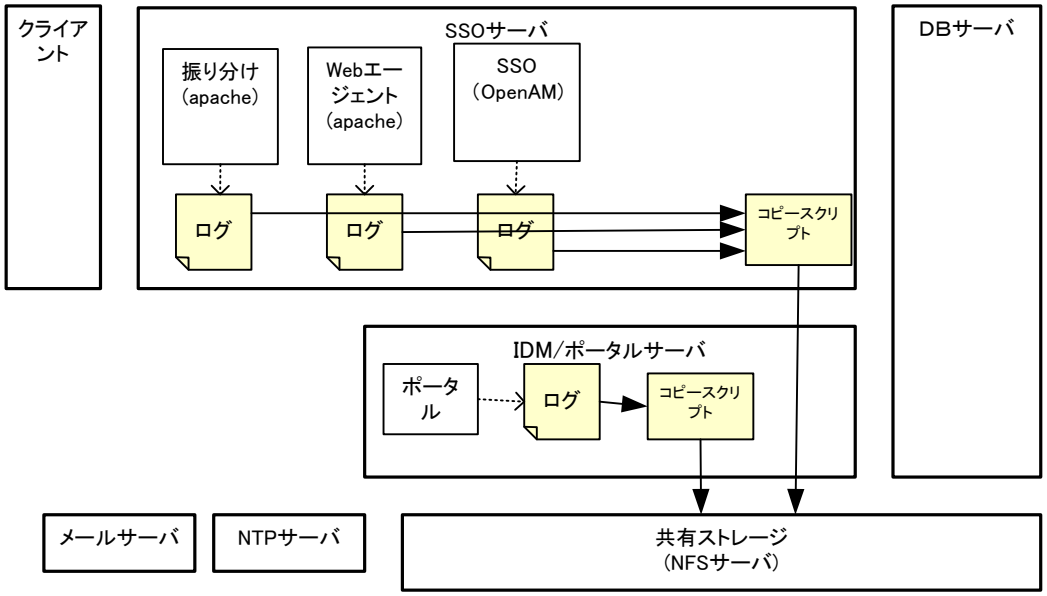


5.1.2. ログ出力

- 以下のミドルのログを出力する
  - Apache
  - Tomcat
  - OpenAM
  - OpenLDAP
  - MySQL
- ログローテーションは以下の通り実施する
  - 1週間以上前のものは削除する。
  - 毎日午前 4:10 にローテーションする
  - 初回のローテーションでは圧縮は行わず、その次のローテーションでログを圧縮する
  - ローテーションする場所は、ログファイルと同じディレクトリとする
  - ログファイルが存在しない場合にエラーを出力しない
- Apache のログローテーションは、Apache を停止し行う。
  - 理由はログローテーション中にアクセスがあった場合は、監査ログが失われるため。

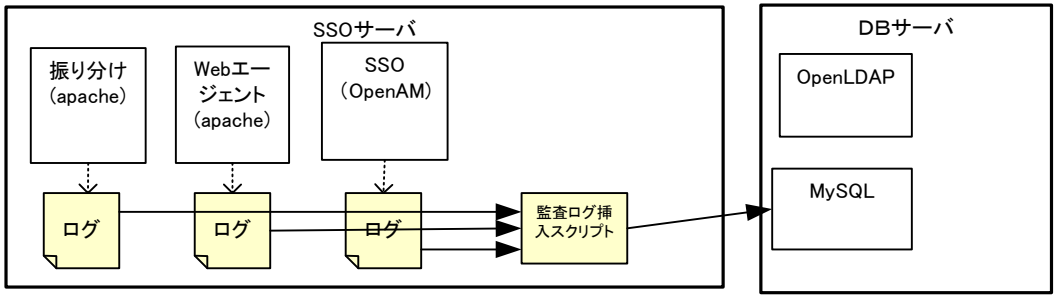
5.1.3. ログ保管

- ログの共有ストレージへのコピー
  - 毎日午前 4:20 にログファイルを共有ストレージへコピーし、18 カ月分保管する。



5.1.4. ログ参照

- SSO のログを MySQL に挿入
  - ポータルから監査ログを見るために、一部のログを MySQL に挿入する。
  - 5:00 に MySQL への挿入を行う。
  - MySQL へ挿入するログの一覧は、表 1 監査対象一覧参照



5.1.5. 稼働統計情報取得

- sar コマンドにより OS の CPU 使用率、メモリ使用率、IO 使用率を定期的に取得する
- netstat コマンドの出力結果を定期的に保存する。

5.1.6. 時刻同期

- 全てのサーバで NTP デーモンを起動し、NTP サーバと時刻同期する。

5.1.7. 常時サービス提供

- すべてのサーバは 24 時間 365 日動作させ続ける。
- ただし、Apache はログローテーションの際に再起動により瞬断するため、そのタイミングではシステムは利用できない。。

5.2. 障害時運用

以下の運用手順書を準備する。

- プロセス起動・停止手順
- MySQL データバックアップ・リストア手順
- OpenLDAP データバックアップ・リストア手順

6. セキュリティ設計

6.1. 監査

監査対象一覧は以下の通り。取得が必要なものは、共有ストレージに移動して、18 カ月分保管する。

表 1 監査対象一覧

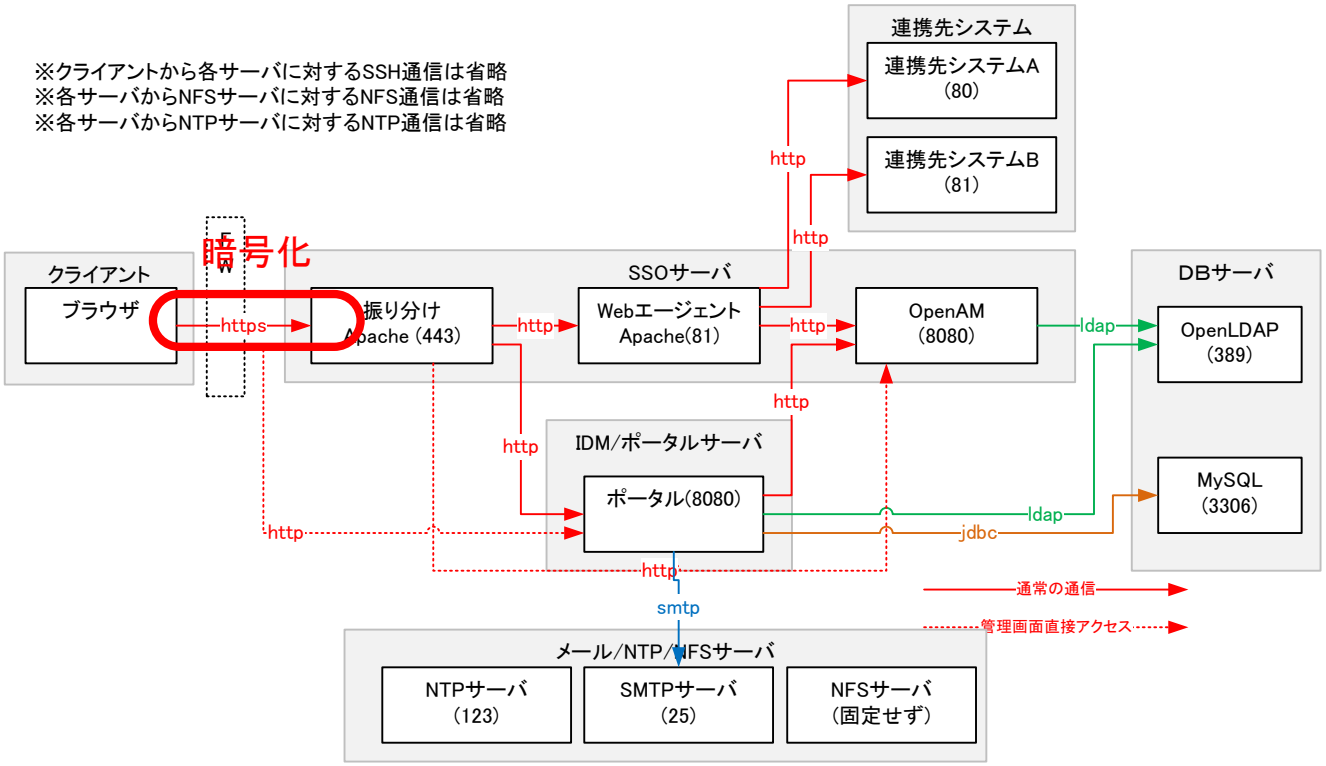
項番	サーバ	プロダクト	種別	対象ログファイル	18 ヶ月保管要非	ポータルからの参照要非（MySQL へ挿入するかどうか）	備考
1	全サーバ共通	OS		messages			
2				secure			
3	SSO サーバ	振り分け(Apache)	Apache ログ	Apache アクセスログ（振り分け用）	○	○	
4				Apache エラーログ（振り分け用）			
5				Apache リライトログ（振り分け用）			
6		Web エージェント(Apache)	Apache ログ	Apache アクセスログ（振り分け＋Web エージェント）	○		
7				Apache エラーログ（振り分け＋Web エージェント）			
8			PolicyAgent ログ	Policy Agent ログ	○	○	
9				Policy Agent デバッグログ			
10		SSO(OpenAM)	OpenAM のイベントやエラーに関するログ	ID-FF イベントログ	○		出力はするが、中身は無い。
11				ID-FF エラーイベントログ			
12				SAML イベントログ	○		本構成では SAML 連携なしのためログは出力されないが、今後の追加要件を考慮して取得対象としておく
13				SAML エラーイベントログ			
14				WSFederation イベントログ	○		出力はするが、中身は無い。
15				WSFederation エラーイベントログ			
16				成功した認証イベントログ	○	○	
17				失敗した認証イベントログ	○	○	
18				コンソールイベントログ	○		
19				コンソールエラーイベントログ			

20				ポリシー許可イベントログ	○		
21				ポリシー許可イベントログ	○		
22				ポリシー拒否イベントログ			
23				SSO の作成と破棄イベントログ			
24			OpenAM 統計ログ	アクセス制御情報			
25				セッション情報			
26				ポリシー情報			
27				キャッシュ情報			
28			Tomcat ログ（Log4J）（OpenAM 用）	tomcat アクセスログ			
29				tomcat 状況ログ			
30				tomcat ログ			
31	IDM ポータルサーバ	OpenStandia ポータル	Apache ログ	Apache アクセスログ	○	○	本構成ではシングル構成で Apache は存在しないためログは出力されない。冗長構成時に Apache を導入した場合は監査対象となる。
32			JBoss ログ	JBossAS サーバログ	○		
				JBossAS バルク登録ログ	○		
33			監査ログ	－ （MySQL に直接書き込み）	○	○	ユーザ、組織属性の更新操作が記録される。
34			サービス監査ログ	－ （MySQL に直接書き込み）	○	○	組織配属、ロール付与の操作が記録される。
35	DB サーバ	MySQL	MySQL ログ	MySQL エラーログ			
36				MySQL スロークエリログ			
37		OpenLDAP	OpenLDAP ログ	OpenLDAP ログ			
38				OpenLDAP 監査ログ			

6.2. データ暗号化

6.2.1. 通信暗号化

クライアント⇔振り分けサーバ間のみ SSL で暗号化する。その他のサーバ間通信はセキュアなネットワークで接続されているため、暗号化しない。



6.2.2. パスワード暗号化

- 代理認証のパスワードは AES（鍵長 128bit）で暗号化する
- ログインパスワードは SSHA1 でハッシュする

6.3. 不正アクセス防止

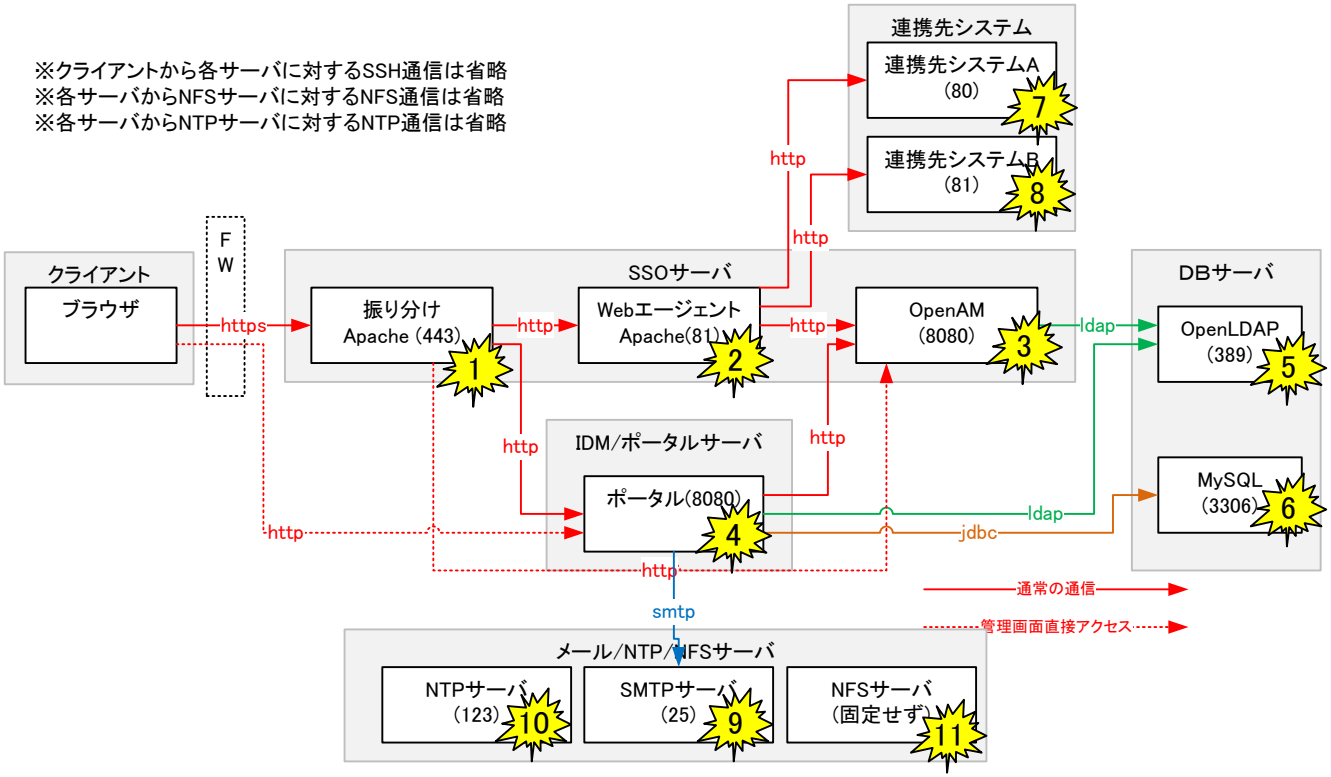
- 本システムでは、不必要な情報の通信を行わないための対策として、サービス提供上不要なポートのクローズや、不要なサービスの無効化を行う。
- OpenAM の管理コンソールには特定の IP からのアクセスのみにする。

7. 耐障害性設計

7.1. 耐ソフトウェア障害

プロセス障害時には、プロセス再起動手順に基づき、再起動を行う。

プロセス毎の障害発生時の影響と対応方法は以下の通り



項番	サーバ	プロセス	影響	エラーの表示箇所	対応方法	備考
1	SSO サーバ	振り分け(Apache)	ログイン、ID 管理等、全ての URL がタイムアウトする。	ブラウザ	問題解消後プロセスの再起動を行う	
2		Web エージェント(Apache)	連携先システムへのアクセスができない(ID 管理およびポータル利用はできる。)	振り分け(apache)	問題解消後プロセスの再起動を行う	

項番	サーバ	プロセス	影響	エラーの表示箇所	対応方法	備考
3		SSO(OpenAM)	ログインできない	振り分け(apache)	問題解消後プロセスの再起動を行う	
			ID管理、ポータル利用ができない	ポータル		
4	IDM/ポータルサーバ	ポータル	ID管理、ポータル利用ができない(シングルサインオンはできる)	振り分け(apache)	問題解消後プロセスの再起動を行う	
5	DB サーバ	OpenLDAP	連携先システムへのログインができない	OpenAM	問題解消後プロセスの再起動を行う	
			ID 管理ができない	ポータル		
6		MySQL	ID管理、ポータル利用ができない	ポータル	問題解消後プロセスの再起動を行う	
7	連携先システムA	連携先システムA	連携先システムAにアクセスするとタイムアウトが発生する	振り分け(apache)	なし。連携先システム担当者に復旧を依頼する	
8	連携先システムB	連携先システムB	連携先システムBにアクセスするとタイムアウトが発生し、ブラウザのエラー画面が出力される	振り分け(apache)	なし。連携先システム担当者に復旧を依頼する	
9	メールサーバ	メールサーバ	即時影響なし。パスワードリセットメールが送信されない。	ポータル	なし。メールサーバ担当者に復旧を依頼する	
10	NTP サーバ	NTP サーバ	即時影響なし。	なし	なし。NTP サーバ担当者に復旧を依頼する	
11	NFS サーバ	NFS サーバ	即時影響なし。	なし	なし。NFS サーバ担当者に復旧を依頼する	

7.2. 耐データ障害

バックアップリストア手順に基づき、データのリストアを行う。

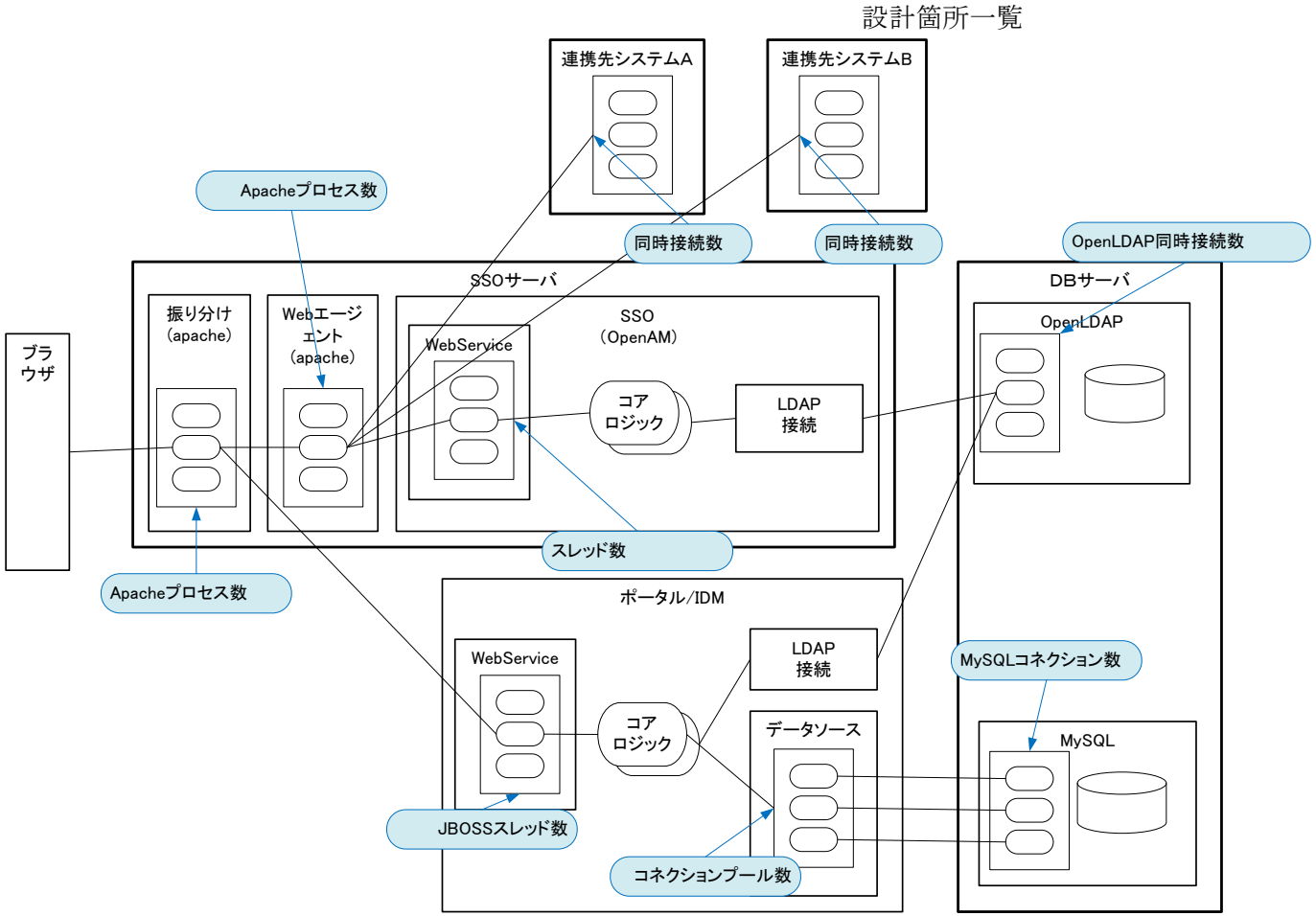
8. 性能設計（サイジング）

8.1. 性能目標

CPU 使用率 80%未満で、全ての性能要件を満たせること

8.2. サイジング

設計箇所一覧



No.	サーバ	設定箇所	設定値		備考
1	SSO サーバ	振り分け(Apache)プロセス数	80		性能要件と過去の実績値により設定。
2		Web エージェント(Apache)プロセス数	80		振り分け(Apache)の MaxClients と合わせる。
3		Tomcat スレッド数	80		振り分け(Apache)の MaxClients と合わせる。
4	ポータルサーバ	JBoss スレッド数(8080 ポート)	20		性能要件と過去の実績値により設定。
5		JBoss スレッド数(8180 ポート)	10		管理者用ポートであり同時アクセス数は少ないため、8080ポートの半分に設定。
6		コネクションプール数(合計)	36	30	lportal コネクションプール数(Liferay) ※JBoss スレッド数と合わせる。
7				5	org001 コネクションプール数(Liferay) ※グループウェアは現時点では使用しないためデフォルトの 5 のままとする。
8				1	lportal コネクションプール数(quartz) ※スケジューラからの DB 接続用に 1 本確保する。
9	DB サーバ	MySQL コネクション数	50		最大想定数+α。バックアップバッチなどの余裕を持たせる。
10		OpenLDAP 同時接続数	1024		ulimit の値となる

9. 拡張設計

9.1. 連携先追加

連携先追加手順書を作成し、その手順に従って追加できるようにする。

9.2. データ量拡張

仮想マシンのディスク容量を増大させることによりデータ量を拡張する。

9.3. 処理性能拡張

仮想マシンのリソース割り当てを増大させることにより、処理性能を拡張する。

10. 移行設計

移行要件なし

11. 端末設計

特に記載する内容は無い

12. 維持管理・サポート方針

ソフトウェアの技術的なサポートについては、OpenStandia の OSS 年間サポートに加入する。
トラブル発生時には、OpenStandia の OSS 年間サポートに問い合わせるが、その際顧客主体でログの取得やプロセスのダウンアップができるように、手順書等を準備する。

13. 開発方針

以下の3つの環境を用意する。

項番		SSO	IDM/ポータル	DB	連携先システム A	連携先システム B	メール	NTP	共有ディスク	備考
1	本番環境									
2	ステージング環境	本番環境と同じソフトウェア構成にする。			本番環境とインターフェースをそろえる。実装方法は問わない。					今回はテンプレートの開発なので準備せず
3	開発環境	本番環境と同じソフトウェア構成にする			ダミーシステムを用いる。		ベンダのメールサーバを用いる	ベンダのメールサーバを用いる	用意しない。ローカルディスクを用いる	今回はテンプレートの開発なので準備せず

14. システムの制約事項

- 連携先システム
  - 連携先システムが ActiveX の場合は追加対応不可
  - ログイン画面のリクエストに対して、HTTP 301 not modified などの body を含まないリクエストを返却するシステムへのSSO連携は対応不可
- システム利用時間
  - 毎日 4:00 にログローテーションを実施するため、システムの応答が一瞬なくなる。

15. 付録

15.1. システム基本情報

15.1.1. ホスト一覧

項番	サーバ名	IP アドレス	ホスト名	備考
1	SSO サーバ	172.105.126.101	v-sso	
2	IDM/ポータルサーバ	172.105.126.102	v-portal	
3	DB サーバ	172.105.126.103	v-db	
4	連携先システム	172.105.126.100	v-app	
5	メール/NTP/NFS サーバ	172.105.126.110	v-mail	

15.1.2. URL 一覧

項番	URL	URL	SSO 対象	備考
1	連携先システム A(ユーザ用)	https://v-openstandia.com/app01/user/	○	
2	連携先システム A(管理者用)	https://v-openstandia.com/app01/	○	
3	連携先システム B(ユーザ用)	https://v-openstandia.com/app02/user/	○	
4	連携先システム B(管理者用)	https://v-openstandia.com/app02/	○	
5	ポータル	https://v-openstandia.com/	○	
6	SSO	https://v-openstandia.com/sso/	○	
7	ポータル管理画面直接ログイン	http://v-openstandia.com:8080/	×	
8	OpenAM 管理画面直接ログイン	https://v-openstandia.com/sso/UI/Login?module=DataStore	×	

15.2. ログファイル一覧

項番	サーバ	ログファイル		
1	全サーバ共通	OS	messages	/var/log/messages
2			secure	/var/log/secure
3	SSO サーバ	Apache ログ	Apache アクセスログ(振り分け用)	/var/log/OpenStandia/apache/access_log
4			Apache エラーログ(振り分け用)	/var/log/OpenStandia/apache/error_log
5			Apache リライトログ(振り分け用)	/var/log/OpenStandia/apache/rewrite.log
6			Apache アクセスログ(振り分け＋Web エージェント)	/var/log/OpenStandia/apache_wa/access_log

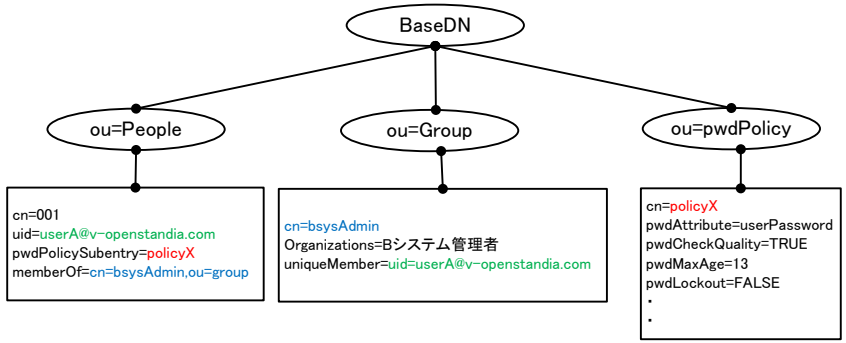
7			Apache エラーログ（振り分け＋Web エージェント）	/var/log/OpenStandia/apache_wa/error_log
8		PolicyAgent ログ	Policy Agent ログ	/var/log/OpenStandia/web_agent/amAgent.log
9			Policy Agent デバッグログ	/var/log/OpenStandia/web_agent/debug/debug.out
10		OpenAM のイベントやエラーに関する ログ	IDFF イベントログ	/var/log/OpenStandia/openam/log/IDFF.access
11			IDFF エラーイベントログ	/var/log/OpenStandia/openam/log/IDFF.error
12			SAML イベントログ	/var/log/OpenStandia/openam/log/SAML2.access
13			SAML エラーイベントログ	/var/log/OpenStandia/openam/log/SAML2.error
14			WSFederation イベントログ	/var/log/OpenStandia/openam/log/WSFederation.access
15			WSFederation エラーイベントログ	/var/log/OpenStandia/openam/log/WSFederation.error
16			成功した認証イベントログ	/var/log/OpenStandia/openam/log/amAuthentication.access
17			失敗した認証イベントログ	/var/log/OpenStandia/openam/log/amAuthentication.error
18			コンソールイベントログ	/var/log/OpenStandia/openam/log/amConsole.access
19			コンソールエラーイベントログ	/var/log/OpenStandia/openam/log/amConsole.error
20			ポリシー許可イベントログ	/var/log/OpenStandia/openam/log/amPolicy.access
21			ポリシー許可イベントログ	/var/log/OpenStandia/openam/log/amPolicyDelegation.access
22			ポリシー拒否イベントログ	/var/log/OpenStandia/openam/log/amPolicy.error
23			SSO の作成と破棄イベントログ	/var/log/OpenStandia/openam/log/amSSO.access
24		OpenAM 統計ログ	アクセス制御情報	/var/log/OpenStandia/openam/stats/Entitlements
25			セッション情報	/var/log/OpenStandia/openam/stats/amMasterSessionTableStats
26			ポリシー情報	/var/log/OpenStandia/openam/stats/amPolicyStats
27			キャッシュ情報	/var/log/OpenStandia/openam/stats/idRepoCacheStat
28		Tomcat ログ（Log4J）（OpenAM 用）	tomcat アクセスログ	/var/log/OpenStandia/tomcat/access_log.txt
29			tomcat 状況ログ	/var/log/OpenStandia/tomcat/catalina.log
30			tomcat ログ	/var/log/OpenStandia/tomcat/tomcat.log
31	IDM ポータルサーバ	JBoss ログ	JBossAS サーバログ	/var/log/OpenStandia/jboss/server.log
34	DB サーバ	MySQL ログ	MySQL エラーログ	/var/log/OpenStandia/mysql-5.5/mysql-error.log
35			MySQL スロークエリログ	/var/log/OpenStandia/mysql-5.5/mysql-slow.log
36		OpenLDAP ログ	OpenLDAP ログ	/var/log/OpenStandia/ldap/ldap.log
37			OpenLDAP 監査ログ	/var/log/OpenStandia/ldap/openldap-auditlog.log

15.1. LDAP のデータ構成

LDAP の構造を以下に示す。

項番	Organization unit	Organization unit 説明	属性値	カスタム	値	備考
1	People	ユーザ情報を格納する。	cn		ユーザ名	
2			uid		ユーザ ID	メールアドレスを想定
3			userPassword		シングルサインオンのパスワード	
4			pwdPolicySubentry		パスワードポリシ名	ou=pwdPolicy の common name と合わせる
5			memberOf		ロール名 書式：cn=(ロール名),ou=group	ou=group の common name と合わせる
6			bsys_id	○	連携先システム B 用の ID	
7			bsys_password	○	連携先システム B 用のパスワード	
8	Group,	ユーザの組織情報を格納する。	cn		組織 ID	
9			description		組織名	
10			uniqueMember		属する組織名	
	Role	ユーザのロール情報を格納する。	cn		ロール ID	
			description		ロール名	
			uniqueMember		属するロール名	
11	pwdPolicy	LDAP の機能にて実現するパスワードポリシー情報を管理する。	cn		パスワードポリシ ID	ポータルで管理、発行される。
12			pwdAttribute			
13			pwdCheckQuality			
14			pwdMaxAge			
15			pwdLockout			

データの例



15.2. LDAP と MySQL と連携先システムのデータ関連図

別紙「【別紙】認証データ関連関連表.xlsx」参照

15.3. URL アクセス制御の説明

OpenAM で可能なアクセスコントロールは、一般利用者がアクセスする URL に対して、アクセス許可/拒否の権限を与えることのみである。細かなアプリケーションレベルのアクセスコントロール機能は無い。

尚、本システムではユーザの権限によってアクセスをコントロールする要件はないので、説明を省略する。

（1） アクセスコントロールの種類

OpenAM のアクセスコントロールには大きく2種類ある。2つを組み合わせてアクセルコントロールを実現する。

- 適用されない URL
- ポリシー

（２） 適用されない URL

適用されない URL とは、認証が不要な URL のことである。

リクエストがポリシーエージェントを通る時、デフォルトでは全ての URL は認証が必要である。
しかし、「お知らせ/案内」画面や「パスワード再発行」画面、画像やスタイルシートなどは、通常認証が不要である。
このような URL に対しては、「適用されない URL」に設定することによって、認証していない状態でも、ログイン画面へ遷移せず、コンテンツを表示する事ができる。

適用されない URL は、ワイルドカード(*)を使用すると、子ディレクトリも対象になる。

例）

https://site01.example.com/contents/* を適用されない URL として設定

→https://site01.example.com/contents/
https://site01.example.com/contents/sub1/
https://site01.example.com/contents/sub2/
https://site01.example.com/contents/sub3/
…
上記がすべて適用されない URL となる。

そのため、**親ディレクトリが適用されない URL で、子ディレクトリは認証が必要な URL という設定はできない。**

NG の例）

https://site01.example.com/sub1/*	→ **適用されない URL**
https://site01.example.com/sub1/private/	→ **認証が必要な URL**

適用されない URL はポリシーエージェント(＝連携サービス)単位に OpenAM 管理コンソールより設定を行う。
また「適用されない URL」は、「反転」オプションを有効にすることで「適用される(＝認証が必要な)URL」となる。

（３） ポリシー

ポリシーとは、認証が必要な URL に対して、認証済みの一般利用者のアクセスを許可/拒否する権限を定義する事である。

ポリシーを設定しない限り、認証済みの一般利用者のアクセスは拒否される。この場合、HTTP ステータス 403（Forbidden）を返却する。
ポリシーも適用されない URL と同様に URL 単位で設定し、ワイルドカード（*）で指定できる。ワイルドカードを指定すると子ディレクトリも対象になる。

例）

```
https://site01.example.com/contents/*        を許可 URL として設定

→https://site01.example.com/contents/
  https://site01.example.com/contents/sub1/
  https://site01.example.com/contents/sub2/
  https://site01.example.com/contents/sub3/
  …
  上記がすべて許可される。
```

例）

```
https://site01.example.com/deny/*            →  許可 URL として設定
https://site01.example.com/deny/allow/*      →  拒否 URL として設定

→https://site01.example.com/deny/allow/
  拒否される。
```

許可と拒否は、**親ディレクトリを拒否、子ディレクトリを許可にすることはできない**。

NG の設定例）

```
https://site01.example.com/deny/*            →  拒否
https://site01.example.com/deny/allow/*      →  許可
```

（４） 適用されない URL とポリシーの評価順序

① 適用されない URL が評価される。
② 適用されない URL に該当しなかった場合、ポリシーが評価される。

（５） アクセスコントロール設計時の留意点

アクセスコントロールの設計はメンテナンス性を重視し、下記点に留意する。

- URL は原則フォルダ単位で指定し、ファイル単位では指定しない事。
- ポリシーの内、拒否 URL は原則使用しない事。

（６） アクセスコントロールの設計／設定例

例） 認証不要な URL が少なく、認証が必要な URL が多い場合。
ACL 要件

URL	認可要件
https://site01.example.com/	認証が必要
https://site01.example.com/sub1/	認証が必要
https://site01.example.com/sub2/	認証が必要
https://site01.example.com/public/	認証が不要

設定値

URL	設定値
適用されない URL の反転	無効
適用されない URL	https://site01.example.com/public/*
ポリシーの許可 URL	https://site01.example.com/*

例） 認証不要な URL が多く、認証が必要な URL が少ない場合。
ACL 要件

URL	認可要件
https://site01.example.com/	認証が不要
https://site01.example.com/sub1/	認証が不要
https://site01.example.com/sub2/	認証が不要
https://site01.example.com/private/	認証が必要

設定値

URL	設定値
適用されない URL の反転	有効
適用されない URL	https://site01.example.com/private/*
ポリシーの許可 URL	https://site01.example.com/private/*