2005－ＤＳＭ－37（７）
2005／5／13

CAS によるセキュアな全学認証基盤の構築

梶田 将司† 内藤 久資‡ 小尻 智子† 平野 靖† 間瀬 健二†

名古屋大学情報連携基盤センター† 名古屋大学多元数理科学研究科‡
〒464-8601 名古屋市千種区不老町 1
E-mail: kajita@nagoya-u.jp

あらまし　本報告では，学内の様々な情報システムが共通に利用できるセキュアな全学認証基盤の実現の１つの方法として，Yale 大学で開発された Central Authentication Service (CAS) を用いて構築した全学認証基盤について述べる．CAS 認証では，ユーザ認証に必要な認証情報を CAS サーバにしか送信しないため，よりセキュアな環境でユーザ認証基盤およびシングルサインオン環境を実現できる．また，HTTP リダイレクション，Cookie などの標準的な Web 技術しか用いられていないため，簡単で軽いシステムである．さらに，LDAP 属性などと併用することにより，強力な Central Authorization Service も容易に実現可能である．本稿では，CAS 認証とその権限管理機能の強化について述べるとともに，平成 17 年度前期開講科目からはじまった全学的な Web 履修申請手続きを対象とした実運用結果について述べる．

キーワード　シングルサインオン，情報セキュリティ，オープンソース，情報基盤，大学ポータル

Developing University-wide Authentication and Authorization Information Infrastructure Using CAS

Shoji Kajita†, Hisashi Naito‡, Tomoko Kojiri†, Yasushi Hirano† and Kenji Mase†

Information Technology Center†, Graduate School of Mathematics‡, Nagoya University
Furo-cho 1, Chikusa-ku, Nagoya 464-8601 JAPAN
E-mail: kajita@nagoya-u.jp

Abstract　This paper describes a university-wide authentication and authorization information infrastructure using Central Authentication Service developed by Yale University and its extention to Central Authorization Service. CAS server is only the server used in authentication so that more secure Single Sign On environment can be attained. Also, CAS is implemented on typical Web standard technology like HTTP redirection and Cookie so that it can be used by quite simple configuration but powerful authentication. Furthermore, CAS can provide not only Central Authentication Service but also Central Authorization Service by using in conjunction with LDAP attributes and so on. In this report, we explain the deployment of CASified Nagoya University Portal for 2005 Web-based registration application, and summarize our experiences.

keyword　Single Sign On, Information Security, Open Source, Information Infrastructure, Institutional Information Portal

1　はじめに

高等教育機関における情報基盤整備は，「学内のコンピュータネットワークの整備」から「大学における教育・研究を支えるアプリケーションの整備」に焦点が移りつつある．しかしながら，コンピュータネットワークとは異なり，アプリケーションは大学における教育・研究の業務プロセスに直結するため，

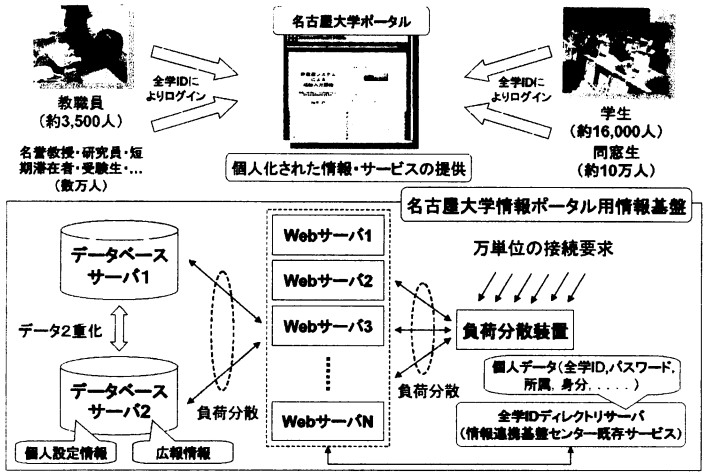

図 1: 名古屋大学ポータルの機器構成.

その活用は，業務プロセスの見直しと IT による業務の効率化という「アカデミックリエンジニアリング」なしには進まない．このことは，WebCT に代表されるコース管理システムの活用が進む北米と比較して，我が国の大学にはなかなか導入が進まない状況 [1] からも伺える．

しかしながら，コンピュータネットワークがそうであったように，アプリケーションの場合においても「個別対応から基盤対応へ」という流れができると考えられる．アプリケーションを実現する仕組みは，最近はほとんどのものが Web 技術であり，大学のような万単位のユーザを対象とする場合，図 1 のような負荷分散や高可用性のある機器構成を取る場合が多い．この構成は，アプリケーション間で共有できるため，まず，ここで基盤対応の可能性が考えられる．

また，最近では，LDAP (Light weight Directory Access Protocol) サーバや Kerberos サーバなどのディレクトリサーバのように，全学レベルでユーザ情報が共有され，情報基盤として運用されるようになってきている．しかしながら，これらの全学レベルでの認証システムで，ユーザ ID とパスワードを共通化することができるが，ユーザは，アプリケーションごとに認証を行わなければならいとともに，認証情報を取り扱うため，アプリケーション側も HTTPS などの暗号化通信によるセキュリティの強化を行わなければならない．

このように，セキュア環境を容易に実現でき，かつ，一度ユーザ認証するだけで他のアプリケーションへのアクセス可能なシングルサインオン機能も実現できる全学的な統一された情報基盤の整備が求められている．

そこで本稿では，学内の様々な情報システムが共通に利用できるセキュアな全学認証基盤の実現の 1 つの方法として，Yale 大学で開発された Central Authentication Service (CAS) を用いて構築した全学認証基盤について述べる．そして，学部 2 年生から 4 年生を対象にした平成 17 年度前期履修登録手続きを通じて行った CAS 化した名古屋大学ポータルの実運用経験についてまとめる．

2 CAS

2.1 概要

CAS は Yale University ITS Technology & Planning が開発した認証機構で，Web ベースのアプリケーションに対してシングルサインオン環境を実現できる．現在は，Java Architecture Special Interest Group のオフィシャルプロジェクトとして，継続的な開発が進められている．

CAS の特徴をまとめると以下の通りである：

- HTTP リダイレクション，Ticket Granting Cookie，Service Ticket (URL パラメータ) という標準的で一般的な Web 技術を駆使するため，処理が極めて軽く，インストールおよび設定が簡単である．CAS 認証を利用するためには，CAS サーバと CAS 認証用のライブラリを利用して認証を行うことになる．
- CAS サーバは Java Servlet で実現されており，アプリケーションが利用することになる CAS 認証を行うためのライブラリとして Java，PHP，

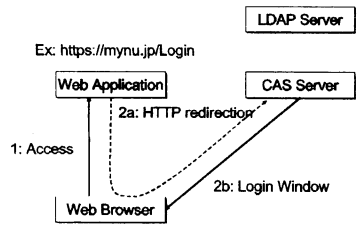

図 2: CAS の動作（その 1）．CAS 認証を経ていないアプリケーションへのアクセスは CAS サーバへ自動的にリダイレクトされ，認証画面が表示される．

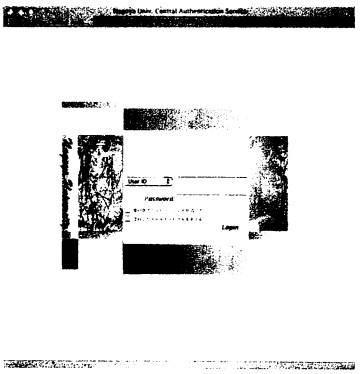

図 3: 名古屋大学 CAS ログイン画面．

Perl, PL/SQL, ASP, Python 用が，また，静的なファイルへのアクセスコントロールが行える Apache mod_cas モジュールや，Zope・Plone 用の CAS ライブラリ，uPortal 用のモジュール，PAM 用モジュールも用意されており，いずれもオープンソースで公開されている．

- 認証に必要なユーザ ID とパスワードは，CAS サーバにしか送られないため，最低限 CAS サーバとエンドユーザ間のみ HTTPS により暗号化通信が行われればよい．
- 豊富な実績．30 を越える大学で利用されている（2005 年 4 月現在）．

2.2 CAS の認証メカニズム

まず，アクセスのために CAS 認証を必要とするアプリケーションに，ユーザが初めてアクセスした場合の動作を説明する．

2.2.1 CAS 認証 (その 1)

ユーザはアクセスしたいアプリケーションの URL を指定 (例えば，`https://mynu.jp/Login`) し，Web ブラウザでアクセスする (図 2 参照)．CAS 認証を経ていない場合，アプリケーション側は CAS サーバへ HTTP リダイレクション機能を使ってアクセスを転送する．その際，`service` パラメータを用いて，CAS サーバが認証すべきサービスを伝える．CAS サーバは，Web ブラウザに保存されている Ticket Granting Cookie (TGC) を確認し，TGC がない場合は，ユーザ認証がまだ終わってないと判断，認証画面（図 3 参照）を表示する．

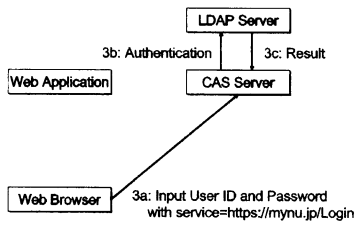

図 4: CAS の動作（その 2）．入力されたユーザ ID とパスワードは，認証源に問い合わされ認証結果を得る．

2.2.2 CAS 認証 (その 2)

ユーザは認証画面で，ユーザ ID とパスワードを入力し送信する (図 4 参照)．入力された認証情報は，LDAP などの認証源に問い合わされ，認証結果を得る．認証源として， LDAP, NIS, RDBMS, 通常のファイルなどを利用可能な CAS Generic Handler を ESUP-Portail consortium が提供している [3].

2.2.3 CAS 認証 (その 3)

ユーザが正しく認証されると，CAS サーバは Web ブラウザに対してセキュア属性[1]をつけた TGC を発行するとともに，URL パラメータ `ticket` に

```
ticket=ST-415240-RGhNbrthiZKezNr9AA7t
```

のような Service Ticket (ST) をセットし，再度，呼び出されたアプリケーションに HTTP リダイレクトする (図 5 参照)．

2.2.4 CAS 認証 (その 4)

アプリケーションは取得した ST を検証するため，CAS サーバに対して ST を送信する（図 6 参照）．CAS サーバでは，メモリ上の ST 発行情報をもとにチケットの正当性を検証する．この際，名古屋大

[1]HTTPS での接続でなければ Web ブラウザは Web サーバに Cookie 情報を送信しない．

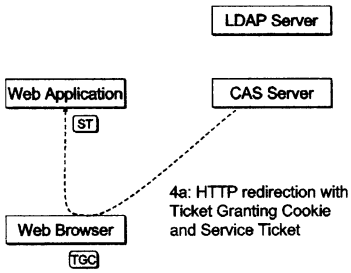

図 5: CAS の動作（その 3）．認証されると，呼び出されたアプリケーションへ自動手的にリダイレクトされる．その際，TGC と ST が発行される．

学の CAS サーバでは，LDAP 属性や時間などによるアクセス制限を課すことも容易に実現できるようにした．この CAS におけるアクセス管理については，次節で詳細に述べる．

アクセスの正当性が確認されると，ユーザ ID がアプリケーション側に通知され，その情報に基づいてアプリケーションはユーザにサービスを提供する (図 7 参照)．

3 CAS による権限管理

我々は CAS に対して，以下の意味での権限管理機構を導入した:

- ST の Validation 要求を受けた際に，アクセスを行おうとするユーザが該当の URL に対するアクセス権を持つかどうかを判断．
- アクセス権がある場合には，CAS クライアントであるアプリケーションに送信するユーザデータベースの属性値を任意に設定可能．

この権限管理機構によるアクセス制限のリストは外部データベースで設定し，CAS の起動時および起動後の任意の時点で設定可能である．このアクセス制限リストを **CAS Access Control List (CAS-ACL)** と呼ぶこととする．今回，名古屋大学ポータルでは CAS-ACL は LDAP DIT 内に格納した．

LDAP を利用した場合，CAS-ACL データベース内の各エントリは以下の 3 種類に分類される．

1. 標準的な CAS-ACL エントリ：Service Validation request に対して Authorization を行う CAS-ACL エントリ．`cas-auth-type` の属性値が `basic` となっている．

```
dn: cn=uPortal,ou=uPortal,ou=cas,o=NU
cn: uPortal
description: uPortal
cas-auth-type: basic
cas-attributes: uid,.....
cas-service: https://mynu\.jp/uPortal/.*
cas-allow: (dn=.+,ou=place.?,o=nu)
objectClass: top
objectClass: cas
```

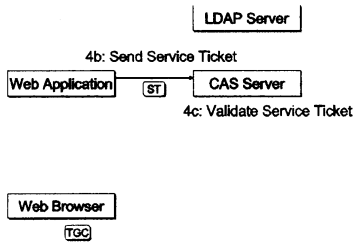

図 6: CAS の動作（その 4）．アプリケーションに送られた ST は CAS サーバに再度送られ正当性が検証される．

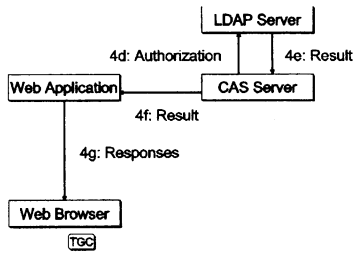

図 7: CAS の動作（その 5）．アクセスしようとしているアプリケーションへの権限があるかどうか LDAP で検証．OK であれば，アプリケーションに必要な LDAP 属性が送信され，それに基づいてアプリケーションはサービスを提供する．

2. basic CAS-ACL エントリのためのマクロエントリ：`cas-auth-type` の属性値が `access_filter` となっている．

```
dn: cn=nagoya-univ-students-b,ou=cas,o=NU
cn: nagoya-univ-students-b
cas-auth-type: access_filter
cas-allow: (&(dn=.+,ou=place.?,o=nn)...
objectClass: top
objectClass: cas
description: Nagoya University User
```

3. CAS-ACL データベースをアップデートする権利を持つユーザを指定する CAS-ACL エントリ：`cas-auth-type` の属性値が `trusted` となっている．

```
dn: ou=uPortal,ou=cas,o=NU
objectClass: top
objectClass: organizationalunit
objectClass: cas
cas-allow: (uid=XXXXX)
ou: uPortal
cn: trusted
description: trust for uPortal
```

この CAS-ACL エントリの構造から，自然に「正規表現で記述された URL のグループ」が構成される．我々は，このグループのことを **CAS Access Control Class (CAS-ACC)** と呼んでいる．すなわち，CAS-ACC とは，アクセス条件と Web アプリケーションに対して送信する属性値が同一となる「正規表現で記述された URL のグループ」である．

4 Web 履修登録による名古屋大学ポータルの実運用

最後に，平成17年度前期履修登録手続きを通じて行ったCAS化した名古屋大学ポータルの実運用経験についてまとめる．

4.1 CAS 化した名古屋大学ポータルの構成

名古屋大学ポータルは，Webサーバ群とデータベースサーバ群で構成される．(図1参照)．Webサーバ群では，Sun Microsystems社のSunFire V210を5台，CAS・LDAPサーバとしてSunFire V480を1台，データベースサーバ群では，SunFire V240 2台とStorEdge 3150FCを用いている．Webサーバ群に対する負荷分散はNortel Networks Alteonが，データベースサーバ群に対しては，Oracle 10g Real Application Clusterが負荷分散を行っている．なお，大学ポータルフレームワークであるuPortalを用いて名古屋大学ポータルを構築している(図3参照)．

Web履修登録処理を行う新教務システムは，SunFire V120 2台およびV210 2台上でOracle Application Serverを動かし，SunFire V240 1台上でOracle 9iのPL/SQLにて独自開発したソフトウェアを用いて実現されている．Oracle ASサーバの負荷分散は，名古屋大学ポータルと同じAlteonで行った．

CAS認証については，uPortalについてはuPortalパッケージおよびJavaクライアントを，新教務システムについてはPL/SQL用CASクライアントを用いて個別にCAS認証を行うように構成した．

4.2 Web 履修登録処理の実運用

名古屋大学では，2年生から4年生までの学部学生約6,500名を対象にWebによる履修登録を2005年3月22日から30日にかけて行った．期間中，学内だけでなく，学外からのアクセスをアクセスも許可し運用した．

4.2.1 限界性能試験

履修登録運用に先立ち，e-Testを用いた負荷実験を行った．その結果を表1に示す．測定は，ボトルネックとなることがあらかじめ分かったデータベースサーバのCPU使用率が85%以上に達した場合とした．表から分かるように，CAS認証を行ったとしても高負荷が予想される履修登録において十分な性能が得られることが分かった．なお，ポータルが履修登録に比べてスループットが悪いのは，ログイン時の処理に時間がかかることが分かっている．

表1: CAS化した名古屋大学ポータルと新教務システムの限界性能．ポータルは，CASが関係するログイン・ログアウトのみを行った．各値は約5分間の平均値．

処理	スループット	レスポンス
内容	[ページ/秒]	[秒]
履修登録	37.5	1.5
集中登録	60.0	1.1
ポータル	17.5	2.4

4.2.2 ユーザの利用状況

CAS認証の回数から見たユーザのアクセス状況を図8に示す．今回は，学外からの利用も許可したため，深夜にわたる利用があったことがよく分かる．

また，ユーザが利用しているWebブラウザは次の通りであった．

```
4199 (57.0%) Windows.XP.MSIE
1757 (23.9%) Windows.2000.Netscape
 757 (10.3%) Windows.98.MSIE
 201 ( 2.7%) Windows.2000.MSIE
 126 ( 1.7%) Linux..Netscape
```

2位のNetscapeは情報メディア教育センターの端末室からの利用を反映している．

4.2.3 アクセス元・経路の状況

アクセスログに基づいて，ユーザのアクセス経路を調査した．経路の同定手順は次の通り：

1. アクセスログに記載されているIPアドレスからFQDNを求める．ただし，10秒以上経ってもレゾルブできない場合はunknownとした．
2. JPIX名古屋に接続されている地域ISP(7社)のドメイン，および，中部テレコミュニケーションに接続されているドメインに属するホストからのアクセスを，JPIX名古屋経由のアクセスと判断した．
3. IPアドレスが133.6.0.0/16の場合は，名古屋大学内からのアクセスと判断した．
4. 上記以外の経路については，すべてSINET経由のアクセスと判断した[2]．

[2]名古屋大学は，JPIX名古屋接続以外はSINETにしか外部経路を持たない．

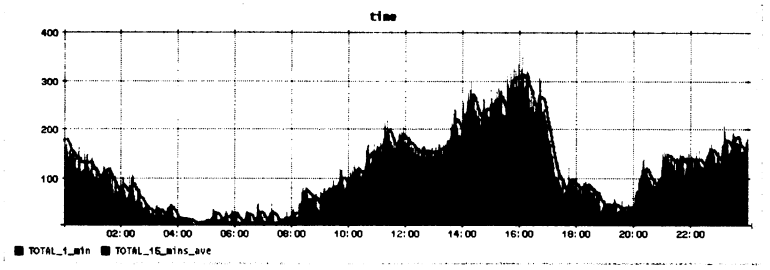

図 8: CAS 認証の回数から見たユーザのアクセス状況．履修登録期間中の平均．18 時から 20 時は保守時間としたためアクセスは少ない．

表 2: 名古屋大学ポータルへのアクセス経路．

アクセス経路	アクセス数
JPIX 名古屋経由	966 (10.2%)
SINET 経由	5,672 (59.9%)
学内から	2,836 (29.9%)
総数	9,474

ただし，学内の端末システムからのアクセスなど，同一ホスト（あるいは，IP アドレス）からのアクセスが想定されるため，アクセス時の全学 ID とホストの FQDN を対にし，アクセス元の総数を同定した．

まず，10 位までのアクセス元を調べたところ，下記のようになった:

```
1922 (26.1%) jp.ac.nagoya-u.media
 984 (13.4%) net.bbtec
 656 ( 8.9%) jp.ne.dion
 641 ( 8.7%) jp.ne.ocn
 403 ( 5.5%) jp.ne.starcat
 229 ( 3.1%) unknown_in_nu
 211 ( 2.9%) jp.ne.so-net
 199 ( 2.7%) jp.ne.aitai
 184 ( 2.5%) jp.ad.mesh
 167 ( 2.3%) jp.or.plala
```

この結果から，今回の履修登録に関しては，1/4 程度のユーザが学内の情報メディア教育センターの端末室からアクセスしていることが分かった．

また，上記の結果から，JPIX 名古屋を経由するアクセス，学内のアクセスの比率，それ以外を表 2 に示す．表から分かるように，10.2% が JPIX 名古屋経由となっている．これは，アクセス元の多くを占める ISP からのアクセスが SINET 経由となっているためであり，直接のピアリングあるいは地域 IX を利用する ISP が増えることで，さらなる改善が期待できる状態であることが分かった．

5 まとめ

本稿では，学内の様々な情報システムが共通に利用できるセキュアな全学認証基盤の実現の 1 つの方法として，CAS を用いて構築した全学認証基盤について述べた．

我々が行った LDAP による権限管理強化は，カナダの Queen's University や他の大学でも似通った拡張が行われており，Central Authentication Service だけでなく，Central Authorization Service のニーズも明確に存在していると言える．コードのオープンソース化など，CAS コミュニティへのフィードバックは必須であろう．

謝辞

本研究は，文部科学省平成 16 年度「知的資産の電子的な保存・活用を支援するソフトウェア技術基盤の構築」研究開発課題「ユビキタス環境下での高等教育機関向けコース管理システム」(研究代表者：間瀬健二)，および，文部科学省科学研究費基盤研究 (A)「地域学術コンソーシアムにおける e-Learning 地域ハブに関する研究」(研究代表者：梶田将司，課題番号：15200054) の助成を受けて実施されている．

参考文献

[1] 日本 WebCT ユーザ会，
http://www.webct.jp/

[2] Yale University ITS Technology & Planning,
http://tp.its.yale.edu/tiki/
tiki-index.php

[3] CAS Generic Hander,
http://esup-casgeneric.sourceforge.net