Overview of Today’s Lecture

- Brief history of user identities
- Single sign-on
- Federated identity model

Popular identity protocols
- SAML
- OpenID
- InfoCard and CardSpace
In the beginning...

... there was almost no interest in creating and managing identities and their security contexts. Why? We lived in a world of mainframes and mini-computers, submitting huge computational jobs through punched cards and printing stacks and stacks of paper on mechanical printers (but only if we were IT professionals or attending University classes at that time). Our identity was nothing more than an identifier, determining who submitted the job and who owned that big amount of paper (usually, printed on the first page of the paper stack).

There was no security context at all in our identities. The user name/password pair was even printed in the punched card set, so that there was absolutely no secrecy involved. However, there was no need for it, especially in the commercial/academic world; except for a few individuals, there was no interest in stealing other people’s jobs (JCL jobs, that is). The only necessary secrets were in the realm of military installations. Identities were used only in the context of a single machine. If you wanted to use another computer, another user name/password pair had to be created, and there was no connection among the identities in the machines that you were allowed to use.

Basically, identities were not used to really identify you. Their only purpose was to generate an identity under which a process was run and the results could be sent to you. There was a very weak connection between you and your digital identity.
A Brief History of Identities

With the advent of distributed computing, network logon became a necessity, and technologies and protocols were specially created to handle those needs. But computers to the users were really a set of resources, presenting the individual a unique view of the resources but, when looking at the local identity management, each workgroup member used exactly the same set of credentials (user name/password) were used to access a set of network resources.

The workgroup concept evolved alongside the network. When file and print resources were present, it was important to have a method to ensure that a printer, no matter which server held the printer queue, I had to identity the local identity database of the member that held the resource; and this individual logic would be responsible for checking if the credentials you used were correct.

This may have been one of the first times when there was a clear relationship between the same set of credentials (user name/password) were used to access a set of network resources.

As can be inferred from the preceding paragraphs, digital identities had to evolve from a single pair of user name/password to a very complex set of protocols that transport lots of user-related claims and attributes. Then came the concept of the network domain. In it, a set of workstations and servers are managed under a central credential management, because there was no need to replicate or clone identities from domain A to domain B if the trust relationship has been previously established. These sets of standards-based protocols are friendly to the Internet infrastructure, allowing the sharing of resources even in the absence of dedicated network links.

New sets of technologies were created and standardized to handle the transmission of user identities among loosely coupled network domains. They are collectively called identity-federation systems: A predefined, cross-platform, standardized set of protocols designed exclusively to transmit user security contexts to allow one network domain to share resources with another network domain. These sets of standards-based protocols are friendly to the Internet infrastructure, allowing the sharing of resources even in the absence of dedicated network links.
Basic Motivating Scenario

- The user is going to travel
- ...or shop
- ...or blog

Tasks
- Sign in for booking flight ticket
- Sign in for booking hotel room
- Sign in for renting a car
Single Sign-On (SSO)

in a **client/server** relationship, single sign-on is a session/user **authentication** process that permits a user to enter one name and password in order to access multiple applications.
Ongoing Identity Crisis

Joe’s Fish Market.Com
Tropical, Fresh Water, Shell Fish, Lobster, Frogs, Whales, Seals, Clams
An Alternative (Web View)
The Non-Web Scenario
Push Toward Unified Identity Management

- Would like to maintain a single identity per user

- That identity act as user credentials for authentication and would be associated with extra user information
  - Name
  - Address
  - email,
  - etc.

- Gets us out of the situation where we have to remember dozens of login/password pairs
Editing User Identity Details
Overview: Federated Identity Model

- The **user** is a person who assumes a particular digital identity to interact with an online network application.

- The **user agent** is a browser or other software application that runs on anything from a PC to a mobile phone to a medical device. A user’s online interactions always take place through an agent, which can passively allow identity information flow or actively mediate it.

- The **service provider (SP)** site is a Web application—such as an expense-reporting application or an open source community—that offloads authentication to a third party, which might also send the SP some user attributes. Because the SP relies on external information, it’s often called a relying party (RP).

- The **identity provider (IdP)** is a Web site that users log in to and that sometimes stores attributes of common interest to share with various SP.
Traditional Identity Management

Institution A

Institution B

Research Projects
Shared Courses
Student Loan Service
Physics Homework Service
Library Provider

= Credentialing / Authentication

= Authorization

= User Credential

"Introduction to Federated Identity Management", John O'Keefe
Federated Identity Concept

Institution A

Institution B

Federation

Research Projects

Shared Courses

Student Loan Service

Physics Homework Service

Library Provider

= Credentialing / Authentication

= Authorization

= User Credential

“Introduction to Federated Identity Management”, John O’Keefe
Example: InCommon Federation

- US Research and Education Federation
  - http://www.incommonfederation.org

- Over 200 participants representing over 4 million users and growing
  - Sponsored partners include the National Science Foundation, the TeraGrid, the National Institutes for Health, EDUCAUSE, the National Student Clearinghouse, and companies offering library databases, human resource systems, and other important services
  - Higher ed. participants include all types of colleges and universities – from the liberal arts to large research institutions

- Members agree to common participation rules and basic practices that allows each to inter-operate with the others

“Introduction to Federated Identity Management”, John O’Keefe
Alice begins her browsing at an SP, such as an investment management site, which she might visit frequently.

Alice wants to access protected resources there, the SP must send an explicit authentication request to Alice’s bank (the IdP).
IdP-Initiated SSO

- IdP, such as a health insurance site, acts as a portal through which Alice accesses various SPs, such as online pharmacies and billing statement aggregators.

- In either case, if Alice’s relationship with an SP predates her IdP relationship, the IdP and the SP accounts must be linked (with her permission) to make SSO successful.
Identity and its Usage is Separate

- Alice can log in once—with one set of credentials—and access multiple Web sites without revealing her credentials to all of them.

- SPs can delegate many account-management tasks (such as password resets) and receive accurate just-in-time user data.

- IdPs can focus on improving authentication methods and adding attractive features to account management interfaces.
Privacy Considerations

- Basic challenge
  - Need to ensure that SPs don’t learn more about the user than absolutely necessary

- Pseudonyms is what’s often used

- However, two basic challenges remain
  - Extra information added to the pseudonym such as postcodes and gender and income can be used to deanonymize the user
  - Multiple SPs can collude and put their information about the user with the same pseudonym together, thereby recovering more information
What Information is personally Identifiable?

Mr. X lives in ZIP code 02138 and was born July 31, 1945

These facts about him were included in an anonymized medical record released to the public

Sounds like Mr. X is pretty anonymous, right?

Latanya Sweeney, a Carnegie Mellon University computer science professor showed in 1997 that this information was enough to pin down Mr. X's more familiar identity -- William Weld, the governor of Massachusetts throughout the 1990s
Gender, ZIP code, and birth date feel anonymous, but Prof. Sweeney was able to identify Governor Weld through them for two reasons.

First, each of these facts about an individual (or other kinds of facts we might not usually think of as identifying) independently narrows down the population, so much so that the combination of (gender, ZIP code, birthdate) was unique for about 87% of the U.S. population.

If you live in the United States, there's an 87% chance that you don't share all three of these attributes with any other U.S. resident.

Second, there may be particular data sources available (Sweeney used a Massachusetts voter registration database) that let people do searches to bootstrap what they know about someone in order to learn more -- including traditional identifiers like name and address.

In a very concrete sense, "anonymized" or "merely demographic" information about people may be neither.

(And a web site that asks "anonymous" users for seemingly trivial information about themselves may be able to use that information to make a unique profile for an individual, or even look up that individual in other databases.)
Architectural Challenges of SSO

- IdP discovery
  - When an SP wants to initial a logon, which IdP do they send the user to?
  - SPs can be bound to a particular IdP
  - Can provide the user with a choice of identity providers

- Or have the user agent decide which identity to use:
  think Android of Facebook phone
User Empowerment

- Focus on user-centric identity
- Give users control about what information is associated with their identity

Privacy:
- Prompt users and require involvement in sharing decisions

Integrity:
- Information about users is not necessarily verified by anyone else, so users can claim to be whoever they want to be
Popular Identity Protocols

- SAML
- OpenID
- InfoCard/CardSpace
Would it make sense for a government entity to be an identity provider?
About NSTIC

The National Strategy for Trusted Identities in Cyberspace (NSTIC) is a White House initiative to work collaboratively with the private sector, advocacy groups, public sector agencies, and other organizations to improve the privacy, security, and convenience of sensitive online transactions.

The Strategy calls for the development of interoperable technology standards and policies — an "Identity Ecosystem" — where individuals, organizations, and underlying infrastructure — such as routers and servers — can be authoritatively authenticated. The goals of the Strategy are to protect individuals, businesses, and public agencies from the high costs of cyber crimes like identity theft and fraud, while simultaneously helping to ensure that the Internet continues to support innovation and a thriving marketplace of products and ideas.
SAML: SAML Assertions

- An assertion contains a packet of security information:
  \[
  \text{<saml:Assertion ...>}
  \]
  ...
  \[
  \text{</saml:Assertion>}
  \]

- How to interpret the assertion:
  Assertion $A$ was issued at time $t$ by issuer $R$
  subject to conditions $C$
Assertion Example

- A typical SAML 1.1 assertion:

```xml
<saml:Assertion
 xmlns:saml="urn:oasis:names:tc:SAML:1.0:assertion"
 MajorVersion="1" MinorVersion="1"
 AssertionID="a75adf55-01d7-40cc-929f-dbd8372ebdfc"
 IssueInstant="2004-12-05T09:22:02Z"
 Issuer="https://idp.example.org/saml">
 <saml:Conditions
   NotBefore="2004-12-05T09:17:02Z"
   NotOnOrAfter="2004-12-05T09:27:02Z"/>
 <!-- insert statement here -->
</saml:Assertion>
```

- The value of the **Issuer** attribute is the unique identifier of the SAML authority.
SAML Statements

- SAML assertions contain statements

- Three types of SAML statements:
  1. Authentication statements
  2. Attribute statements
  3. Authorization decision statements

- Although statements are the “meat” of assertions, the assertion remains the atomic unit of SAML
A typical *authentication statement* asserts:
Subject S authenticated at time \( t \) using authentication method \( m \)

- A `NameIdentifier` refers to subject S
- The `NameIdentifier` has properties:
  - transparent or opaque
  - persistent or transient
In a statement, the SAML Subject is crucial:

```xml
<saml:Subject
xmlns:saml="urn:oasis:names:tc:SAML:1.0:assertion">
  <saml:NameIdentifier Format="urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress"
NameQualifier="https://idp.example.org/saml">
    user@example.org
  </saml:NameIdentifier>
</saml:Subject>
```

In this example, the Format of the NameIdentifier is an emailAddress, a transparent, persistent identifier.

In deployments where privacy is an issue, an opaque, transient identifier is more appropriate.

Unfortunately, SAML 1.1 does not specify such an identifier (but SAML 2.0 does)
Statement Example

- A subject-based authentication statement:

```xml
<saml:AuthenticationStatement
  xmlns:saml="urn:oasis:names:tc:SAML:1.0:assertion"
>
  <saml:AuthenticationInstant>2004-12-05T09:22:00Z</saml:AuthenticationInstant>

  <saml:Subject>
    <saml:NameIdentifier
      Format="urn:oasis:names:tc:SAML:1.1:nameid-format:X509SubjectName"
      NameQualifier="https://idp.ncsa.uiuc.edu/saml">
      CN=GridShib,OU=NCSA,O=UIUC
    </saml:NameIdentifier>
  </saml:Subject>
</saml:AuthenticationStatement>
```

- In this example, we use an X.509 subject DN as a NameIdentifier

- Note also the time and method of authentication
Shibboleth

- First large-scale Federated Security solution
- Secures web sites and web applications
- Implements Security Assertion Markup Language (SAML) standard
- Initially developed for research and higher education
  - Research collaboration
  - Academic information providers
  - Outsourced employee applications
  - Extended user populations
- Open source project
Security Assertions

- Attributes assigned to user accounts
- Represent group affiliation or user privilege
  - No predefined semantics by Shibboleth
  - Semantic agreement among participants
  - Federation and two-party arrangements
- Bundled with resource requests
  - Authenticated by IdP
  - Basis of resource authorization by SP
Shibboleth Web Application SSO

Web Application SSO Details

- Based on SAML Web Browser SSO Profile
- Standard browser request, e.g. GET
- Where-Are-You-From service locates IdP

- User browser redirected to IdP
  - Automated with JavaScript or manually invoked
- IdP specific identity verification
- Digitally signed security assertions
- Browser session enables single sign-on
What is OpenID

- **URL**
  - Unique to user
  - User can claim
  - Use for authentication

- **Single-Sign On**

- **Decentralized**: URL can reside in any domain

- **Anonymous**: URLs (pseudonyms) are used
OpenID In Use

Log in with OpenID

Click your OpenID account provider:

Google  Yahoo!  myOpenID  AOL

Or, manually enter your OpenID URL:

[Input field]

Login

Or, if you don't have an OpenID through any of the above, click here to sign up!

Can't remember your login information?

Why OpenID?

It's a single username and password that allows you to log in to any OpenID-enabled site.

It works on thousands of websites.

It's an open standard.

use your own URL

Want to add OpenID support to your website?

It's as easy as adding two HTML header tags!

see how
May 2005 – OpenID authentication protocol developed by Brad Fitzpatrick

May 2006 – JanRain developed Simple Registration Extension (profile-exchange)

May 2006 – Incorporate XRI support

Jan 2007 – Symantec supports OpenID
OpenID History

- Feb 2007 – Microsoft, AOL supports OpenID
- May 2007 – Sun Microsystem supports OpenID
- June 2007 – OpenID Foundation formed in Oregon
- Jan 2008 – Yahoo! Supports OpenID
- Feb 2008 – Google, IBM, VeriSign, and Yahoo joined OpenID Foundation corporate board
- In January 2009, PayPal joined the OpenID Foundation as a corporate member, followed shortly by Facebook in February
Sites Supporting OpenID

Unique Relying Parties as of Jan 1st 2009
Key Adopters
How OpenID Works

RP – Relaying Party: OpenID Supported Page
OP – OpenID Provider: such as livejournal.com or aol.com

1. User initiates authentication process
2. RP Perform Discover/Normalize identifier
3. Establish an Association (Diffie-Hellman Key Exchange)
4. RP directions User to OP with request
5. OP Authorizes/Deny request
6. OP redirects User to RP with authorization approved/denied
7. RP verifies information + OP sources
Self-Hosting an OpenID

<link rel="openid.server" href="http://www.myopenid.com/server" />
<link rel="openid.delegate" href="http://youraccount.myopenid.com/">
<link rel="openid2.local_id" href="http://youraccount.myopenid.com" />
<link rel="openid2.provider" href="http://www.myopenid.com/server" />
OpenID Scenario (1)

Enter OpenID Supported Page (Relaying Party)
OpenID Scenario (2)

- OpenID Login (http://openid.aol.com/koovaj)
OpenID Scenario (3)

- Redirected to OpenID Provider for auth
OpenID Scenario (4)

- Redirect to Relaying Party (granted/denied)
Phishing is a Challenge

1. Send user's URL
2. Redirect user to get token from Evil Scooper
3. Get token from ES
4. Scooper contacts OP and creates a visual replica at ES to dupe user
5. User posts Creds to Evil Scooper
MS Passport: Fake Merchant Attack

- Same as phishing issues we saw before
  - Bob = Passport user
  - Mallory = Attacker of Malicious party

- **Assumption**: Bob get accustomed to using passport and trust the security of the passport server
How to Attack?

1. Mallory sets up a phony web

2. Mallory gets a certificate for a web site, called pasport.com. And Mallory sets up his web site which is exactly the same as a real passport.com.

3. So Bob want to buy something in Mallory’s shop, click sign-in, the server creates a redirect to Mallory’s pasport.com. Bob is in the habit of filling his Email Address and Password

4. After that, Mallory has got Bob’s valid authentication information, and he can go to online shop, use Bob’s wallet service on behalf of Bob
Attacks on MS Passport

- Fake merchant attack
- DNS poisoning attack
- Client-side Cookie-based attack
Windows CardSpace

- Windows CardSpace is a piece of client software that enables users to provide their digital identity to online services in a simple, secure and trusted way.
CardSpace Environment

- Runs under separate desktop and restricted account
- Isolates CardSpace runtime from Windows desktop
- Deters hacking attempts by user-mode processes
CardSpace Cards

- Contains claims about my identity that I assert
- Not corroborated
- Stored locally
- Signed and encrypted to prevent replay attacks

- Provided by banks, stores, government, clubs, etc
- Locally stored cards contain metadata only!
- Data stored by Identity Provider and obtained only when card submitted
- Users can’t edit claims
- Can be protected by various means (Username/Password, Kerberos, SmartCard etc)
The Identity Selector

Easier:
No usernames
No passwords

Consistent:
Same UI

Safer:
Avoids Phishing
Multi-factor authentication
The Typical Logon Process

1. Login to identity provider
2. Token issued to client
3. Token sent to service provider
4. Token validated with identity provider
5. Output sent to client
The CardSpace Logon Process

1. Service Provider Requests Identity
2. CardSpace Identity Selector pops up
3. Token is built by Identity Selector (with Identity Provider)
4. Token sent to client
5. Output sent to client
# CardSpace Versus OpenID/Passport

<table>
<thead>
<tr>
<th>Cardspace</th>
<th>Open ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client side prompt (IE support/FireFox <a href="#">community code</a>)</td>
<td>HTML Form</td>
</tr>
<tr>
<td>Common User Experience</td>
<td>Experience varies between Identity Providers</td>
</tr>
<tr>
<td>Simpler Login</td>
<td>Redirection / Site Bounce</td>
</tr>
<tr>
<td>Requires EV SSL</td>
<td>No SSL required</td>
</tr>
</tbody>
</table>
Requesting a CardSpace InfoCard

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>Sample 1</title>
</head>
<body>
  <form id="form1" method="post" action="login1.aspx">
    <button type="submit">Click here to sign in with your Information Card</button>
    <object type="application/x-informationcard" name="xmlToken">
      <param name="tokenType" value="urn:oasis:names:tc:SAML:1.0:assertion" />
      <param name="issuer" value="http://schemas.xmlsoap.org/ws/2005/05/identity/issuer/self" />
    </object>
  </form>
</body>
</html>
CardSpace Identity Selector

Do you want to send a card to this site?

Site information:

Fabrikam
www.fabrikam.com
Fabrikam
Redmond, Washington, US
View privacy statement
Cards that are sent to this site may be sent to the site's designated agents.

Site information verified by:

A. Delta Corporation
Adatum

Yes, choose a card to send
No, return to the site
Creating a Personal Card

![Image of Windows CardSpace window with a card that has not been sent to the site]

- You have not sent this card to the site. Review the card before you send it.

**Card data that will be sent to this site:**
- First Name: Frank
- Last Name: Lee
- Email Address: flee@contoso.com
- Site-specific card ID: UZM-49FD-ZY2

**Recent card history (not sent):**
This card has not been used before.

**Additional card details (not sent):**
- Created On: 9/12/2006
Locking A Card

A PIN can use any combination of letters, numbers, and symbols. A minimum of four characters is required. A minimum of eight characters is recommended.

New PIN:

Confirm new PIN:

Personal Card
Summary

- Brief history of user identities
- Single sign-on
- Federated identity model

- Popular identity protocols
  - SAML
  - OpenID
  - InfoCard and CardSpace