RSAConference2015

San Francisco | April 20-24 | Moscone Center

CHANGE

Challenge today's security thinking

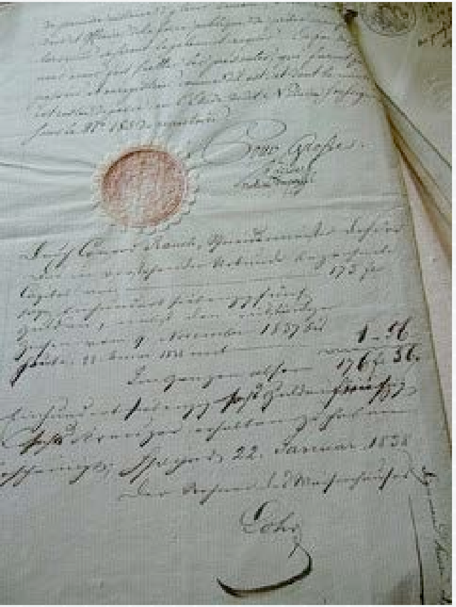
SESSION ID: IDY-R01

Standards for Exchange of Identification Context between Federated Parties

Pamela Dingle

Principal Technical Architect
Ping Identity
@pamelarosiedee

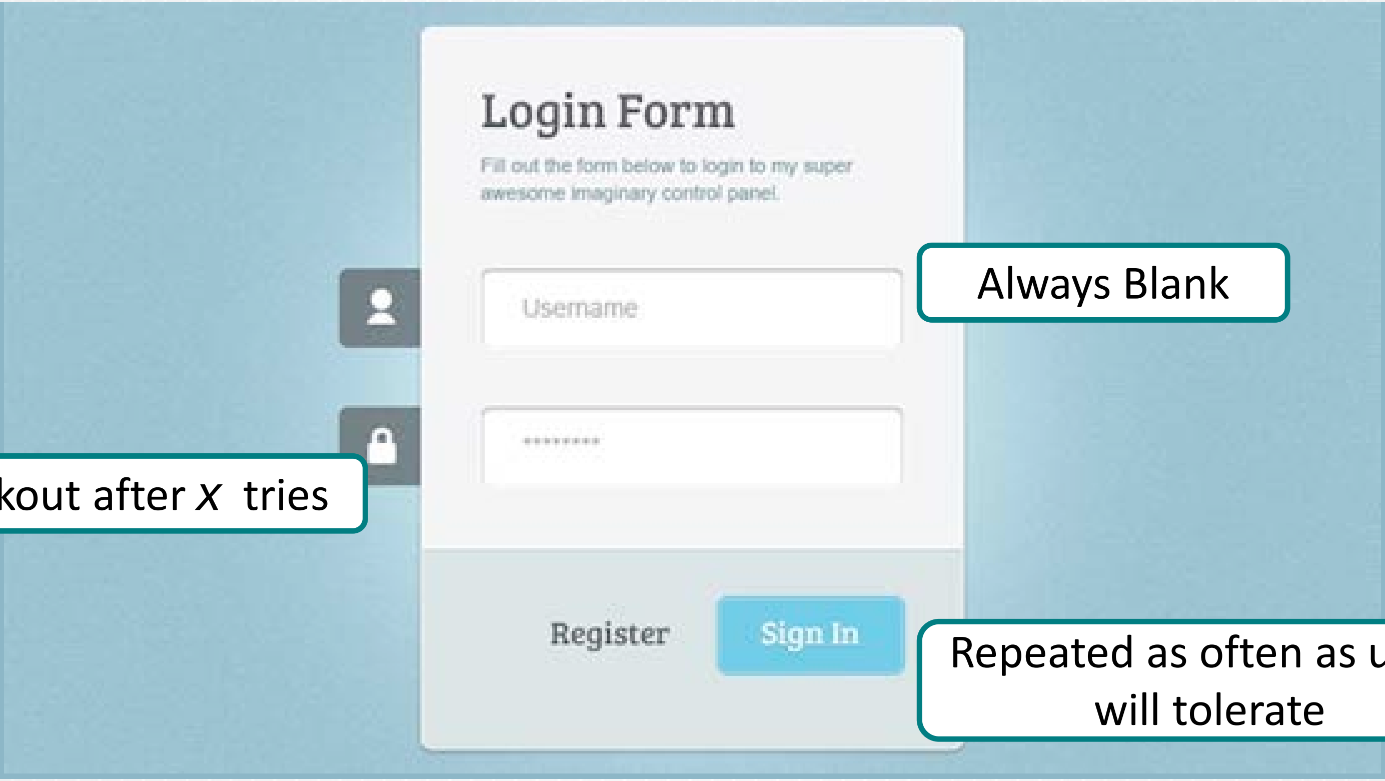
Quick Agenda



- The Stateless Present
  - IDP Discovery, Strangers and Cookies
- Changing Outlooks
- Choosers and Login Hints
  - login_hint vs. id_token_hint
- Why is it Relevant and How can it be Used
- Future of Federated Context Sharing
- Recommendations / How to Apply

https://www.flickr.com/photos/tusnelda/6141350136/

The Stateless Present



Login Form

Fill out the form below to login to my super awesome imaginary control panel.

Always Blank

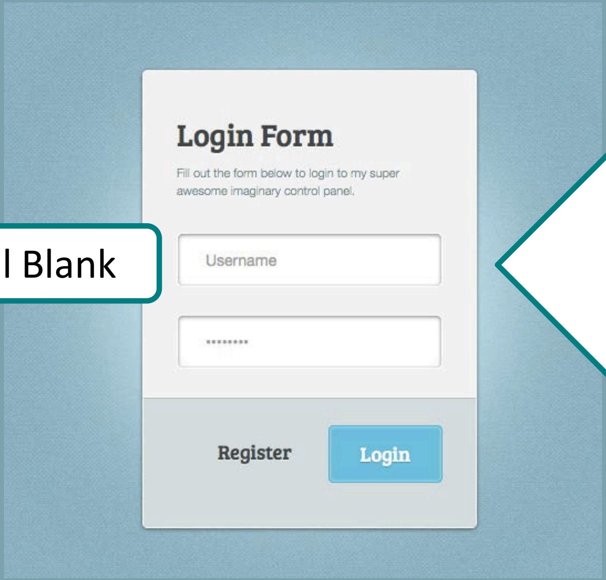
Lockout after *x* tries

Repeated as often as users will tolerate

Source: quickwebresources.com

IDP Discovery Often Precedes Federation

Still Blank



Login Form

Fill out the form below to login to my super awesome imaginary control panel.

Username

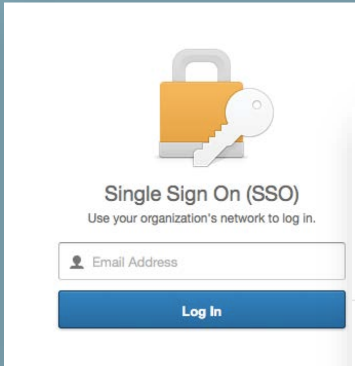
••••••••

Register Login

Identity Provider

Authentication Request

Many / Private Partners



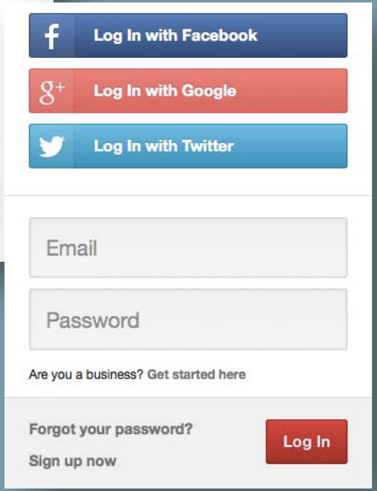
Single Sign On (SSO)

Use your organization's network to log in.

Email Address

Log In

Few / Public Partners



Log In with Facebook

Log In with Google

Log In with Twitter

Email

Password

Are you a business? Get started here

Forgot your password?

Sign up now

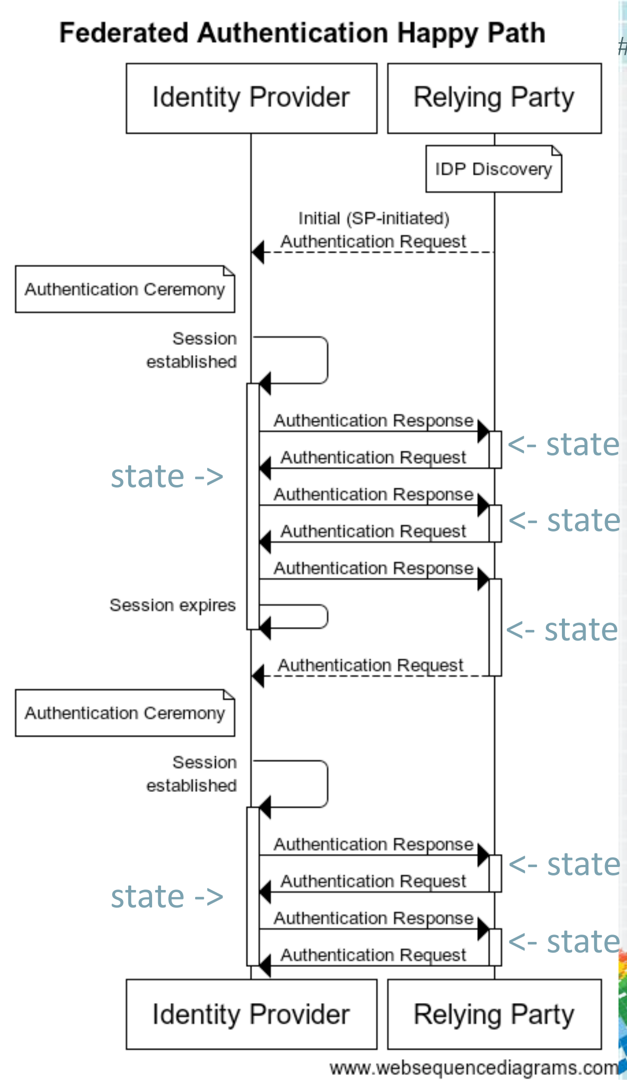
Log In

Relying Party

Source: quickwebresources.com, pinterest, box

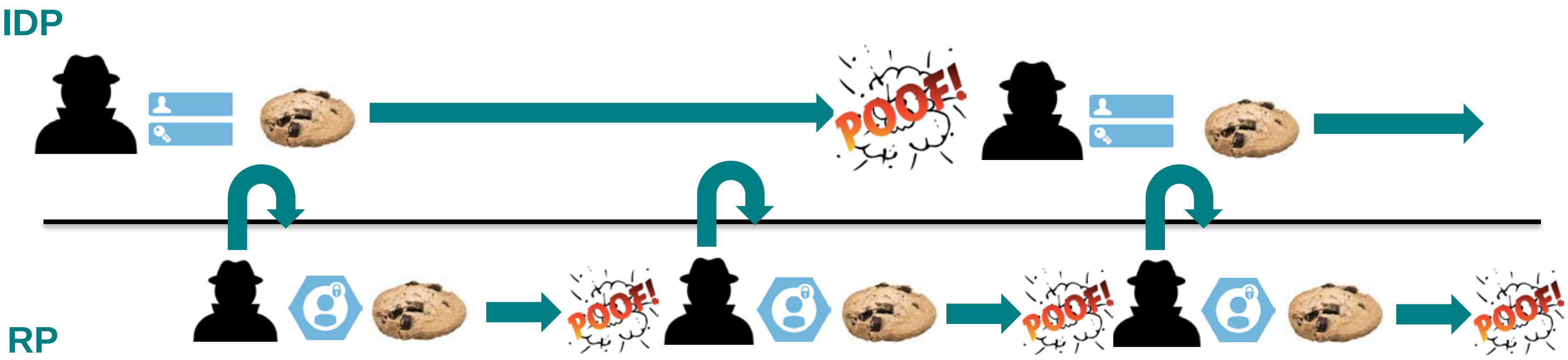
But Context is Rarely Linked

- Today, most sessions are independently established
- Some state may be preserved at a relying party domain (ie via cookie) but is not shared
- Height of state preservation today at IDP: “Remember Me” button
- Few correlate state across partners



Current Practice: Strangers and Cookies

- Looking at a user’s interaction with a single resource, it is a series of tests given to strangers, separated by cookie lifetimes



Are we stuck here?

- Why are we strangers on corporate devices that we exclusively use every day
- How can users help systems to identify accounts
- Can federated domains collaborate in a standardized way?
- What trends could be pushing us in new directions?

https://www.flickr.com/photos/bensonkua/2754312951

Authentication Attitudes are Changing

- Authentication architectures have been historically based on the sentiment of only accepting information that can be validated, with the idea that if you receive it you can trust it.
- Password reuse is a major breach cause
  - Databases of username/credential combinations that could validate, collaboratively assembled and maintained, preying on password reuse
- The entire industry is moving towards a different paradigm: more data, of lower assurance, trusted less individually but evaluated in concert and over time

Photo: W_Minshull, https://www.flickr.com/photos/23950335@N07/6034683535

Usability Attitudes are Changing

- Device portability is changing the usability landscape
  - Frequency of authentication
  - Limited data input options
  - User-not-present use cases (notifications, alerts,)
  - When a device is public & stationary, it is socially acceptable for anyone to login. When a device is portable, it <u>belongs</u> to somebody.
  - Many have experienced device loss first-hand
  - Highly publicized photo theft instances
- Reduction of typing a <u>critical</u> consideration for app developers

New Identities are in Play

- Client Identity:
  - Scoped authorization frameworks like OAuth 2.0 (RFC 6749/50) frame everything in terms of a requesting client.
  - OpenID Connect discovery & dynamic registration specs give the potential to assign a different identifier to every instantiation of software separately.
- Device Identity:
  - Heavy work is occurring to securely probe & understand the ‘posture’ of the device on which the software is running.
    - Is it “trusted”? What is the relationship with the user?
    - Is there malware?

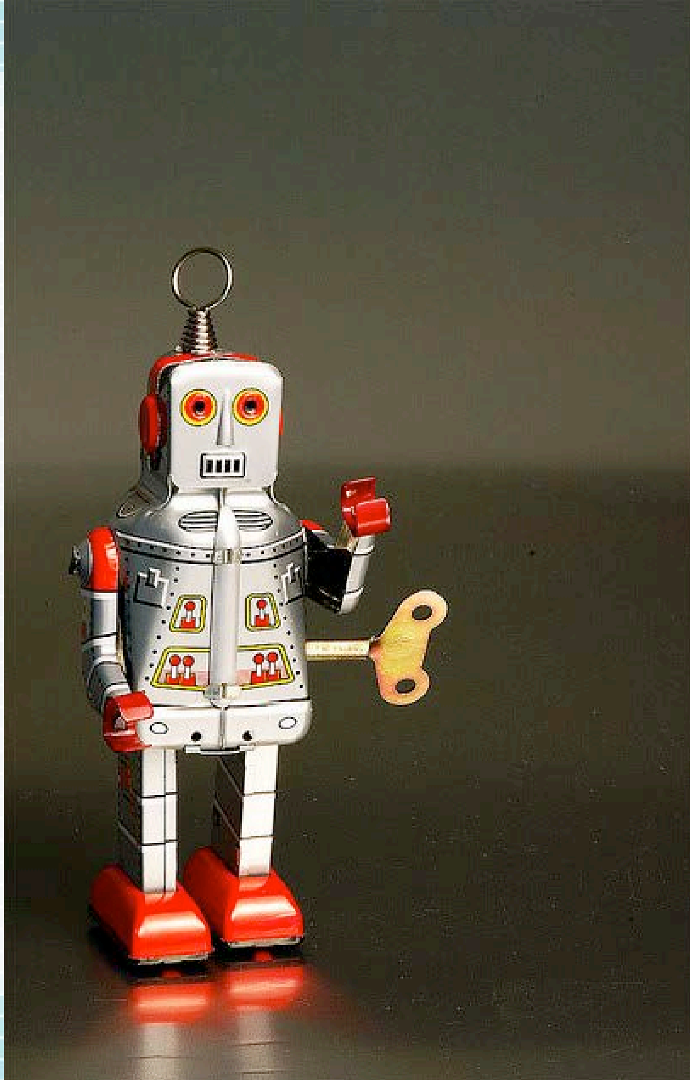
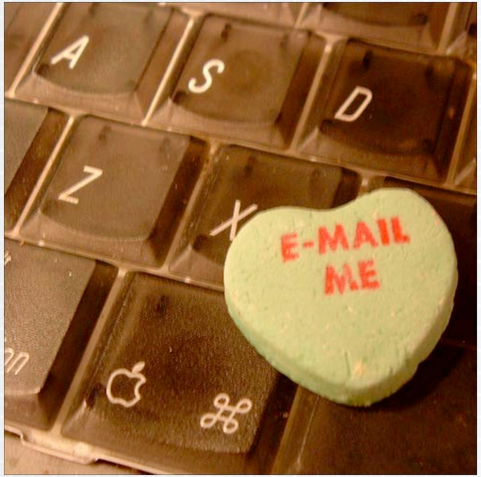


Photo by Mark Strozier https://www.flickr.com/photos/r80o/39304743

Even Identifiers are changing



https://www.flickr.com/photos/idogcow/391609724

- Usernames common in Enterprise still
  - But are often related to or derivable from email
- Cloud Apps moving towards email as login ID (consumer and Enterprise)
  - Upside
    - Built in global uniqueness
    - Easy to remember
  - Downside
    - Global correlation key

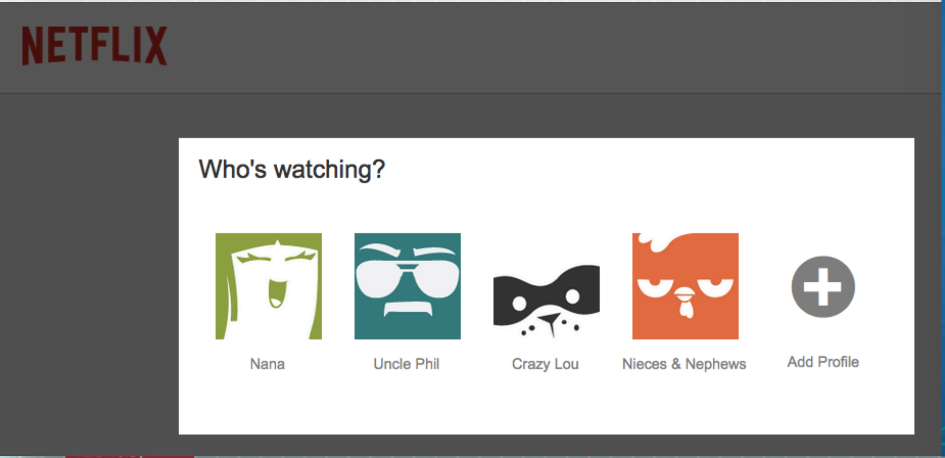
Now standardized: the “Login Hint”

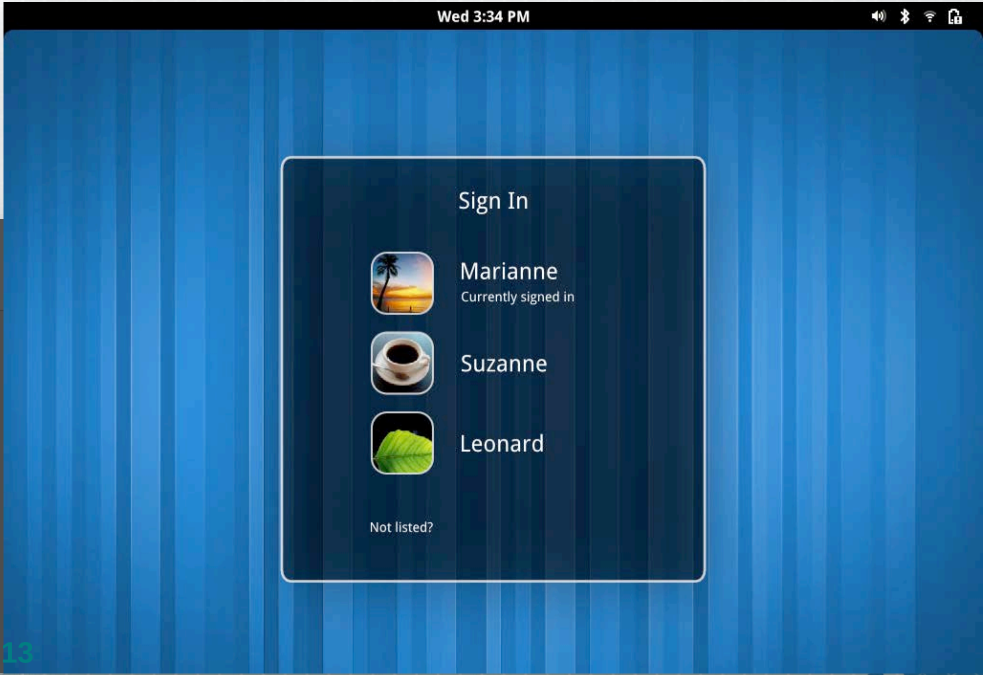
- A guess on the part of a Federated Relying Party as to the identity of the user sent to the Identity Provider
  - Hints can be determined by:
    - Prompting the user
    - Referencing a recently expired RP session
    - Caching the last IDP assertion sent to this client
- Genesis: OpenID 2.0 ‘user claimed identifier’
  - Blazed trails around globally unique identifier usability
  - OpenID Connect & Account Chooser take this idea one step further
- Think of it as: user-provided context

https://www.flickr.com/photos/moofbong/4220715069

Login Hints are used in Choosers

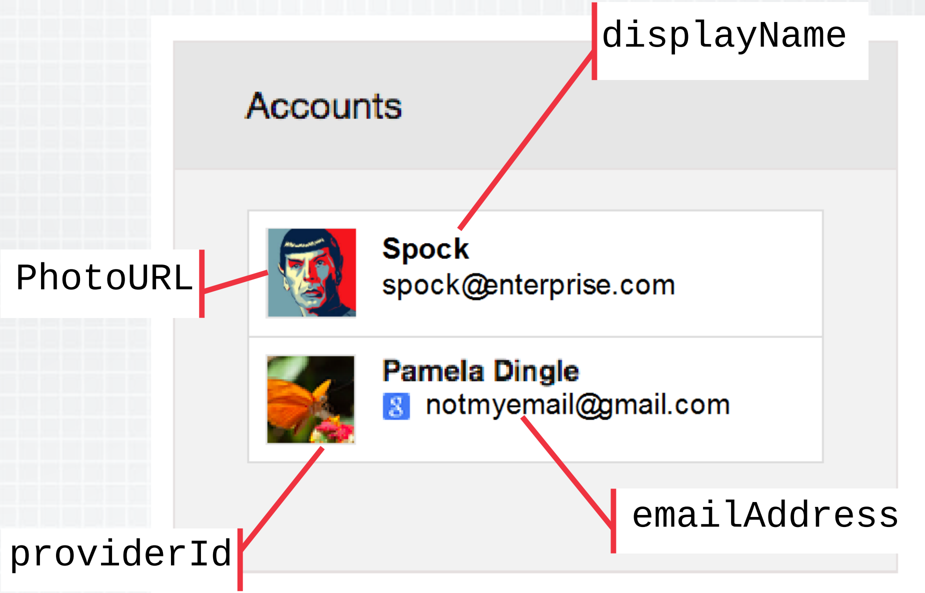
- Choosers are graphical user login menus meant to make logging in easier the 2nd time a user interacts
  - Pretty but proprietary
  - Do not authenticate, only refer





Chooser Standardization

- Account Chooser specs standardizes data and javascript API for choosers
  - Goal is reuse of chooser information across websites (with and without federation) for login and registration
  - Try it at: http://hipstabank.com
  - Spec at: http://openid.net/ac
- Stored: 4 pieces of information



displayName

Accounts

PhotoURL

Spock
spock@enterprise.com

Pamela Dingle
notmyemail@gmail.com

emailAddress

providerId

Standards for Communicating Login Hints

- OpenID Connect Simple Login Hint

```
HTTP/1.1 302 Found Location: https://server.example.com/authorize?
  response_type=code
  &scope=openid%20profile%20email
  &client_id=s6BhdRkqt3
  &state=af0ifjsldkj
  &redirect_uri=https%3A%2F%2Fclient.example.org%2Fcb
  &login_hint=spock%40enterprise.com
```

Use of Login Hints

- Bootstrapping
  - When you hit a “cold” RP scenario where no context is known, prompting the user with an account chooser gives the relying party the ability to leverage pre-stored account credentials (with consent of the user)
- Continuation
  - In a “hot” RP scenario, where a session has previously existed, sending a new request containing the last used IDP assertion or identifier could communicate valuable context, both improving security and usability
- Context Switching
  - If the relying party supports the “log in as another user” feature from within a session, the account chooser is an easy way to allow quick switches.
- Note that both Bootstrapping and Context Switching are also useful in non-Federated contexts.

Triggering a Chooser using AC Spec

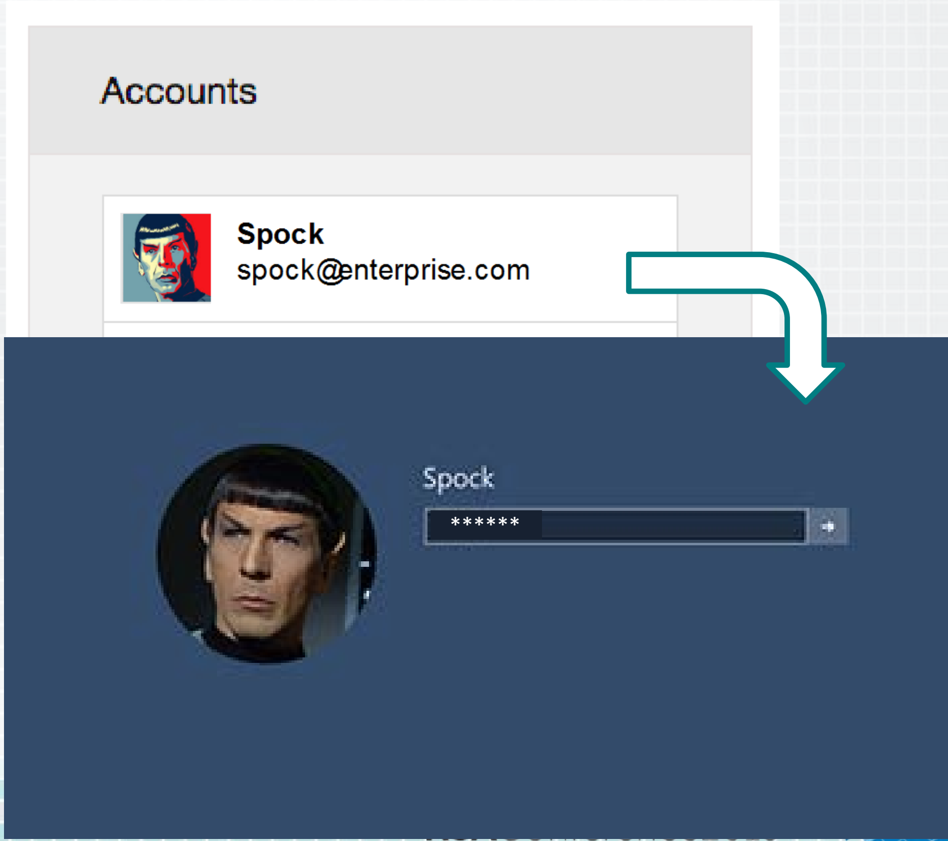
```
<html>
 <head>
   <script type="text/javascript"
     src="https://www.accountchooser.com/ac.js" />
   <script type="text/javascript">
          accountchooser.CONFIG={
           loginUrl: "utils/mysitelogin",
                       signupUrl: “utils/mysignup”,
           mode: "login",
           siteEmailId: "form_username",
           sitePasswordId: "form_password" };
   </script>
 </head>
 <body>
   <form>
     <input id="form_username" type="text” />
     <input id="form_password" type="password" />
     <input id="submit" type="submit">Login</input>
   </form>
```

Redirects to signupUrl if account doesn’t exist

Populates form and sets focus in non-federated case

What Does this Get You? Or an Attacker?

- What does the User get?
  - Less Typing! More Usability!
- What could an Attacker get?
  - Not much. It is garbage in, garbage out.
  - Some 1st factors problematic – but that is true even without hints
- What does the Identity Infrastructure get?
  - Advance notice to start running fraud/risk evaluation!
  - Establishment of ceremony & behavior



Accounts

Spock
spock@enterprise.com

Spock
******

Standards for Communicating Hints

```
HTTP/1.1 302 Found Location: https://server.example.com/authorize?
  response_type=code
  &scope=openid%20profile%20email
  &client_id=s6BhdRkqt3
  &state=af0ifjsldkj
  &redirect_uri=https%3A%2F%2Fclient.example.org%2Fcb
 &id_token_hint=eyJ0…NiJ9.ey1c…ifX0.DeWt4Qu…ZXso
```

Previously received assertion “id_token” sent back to IDP during authentication request

Full of state goodness

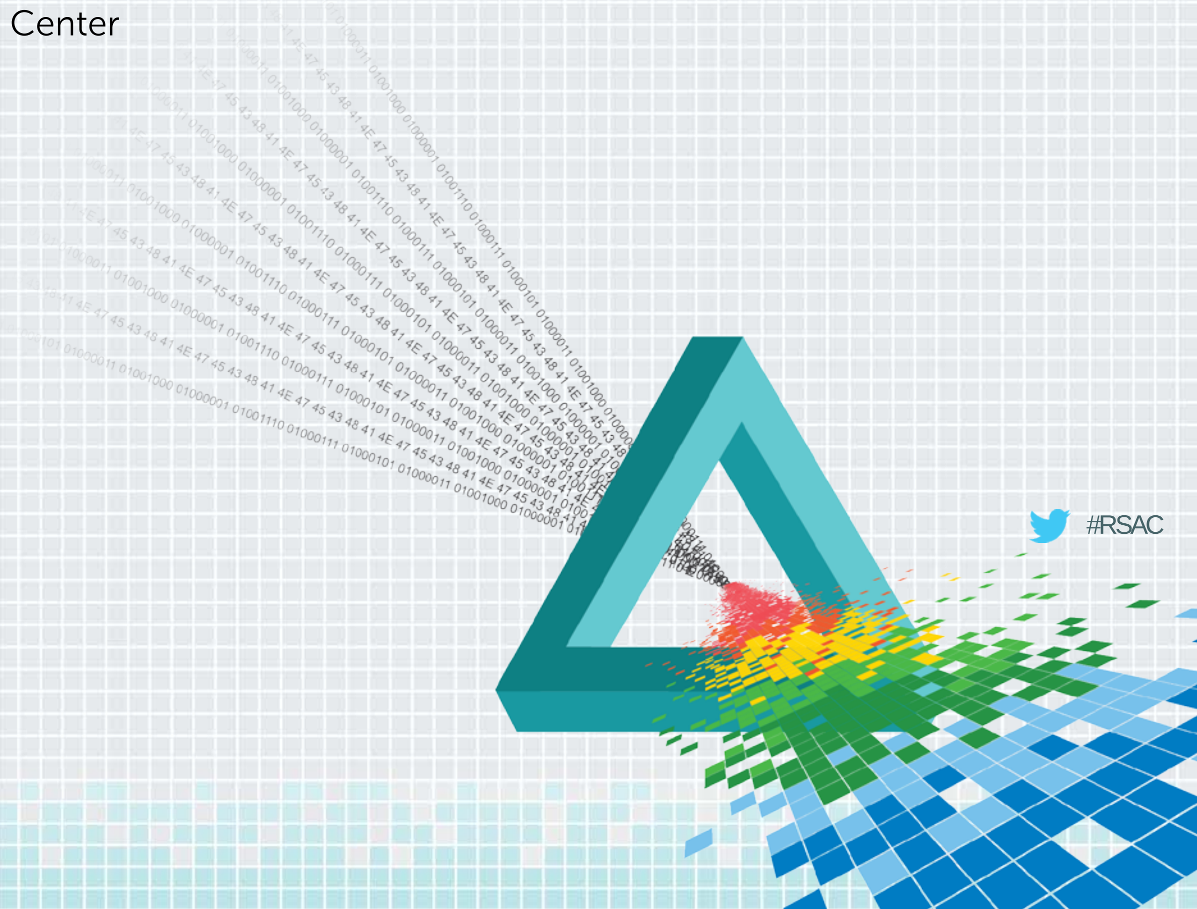
```
{
"iss": "https://server.example.com",
"sub": "24400320",
"aud": "s6BhdRkqt3",
"nonce": "n-0S6_WzA2Mj",
"exp": 1311281970,
"iat": 1311280970,
"auth_time": 1311280969,
"acr": "urn:mace:incommon:iap:silver"
}
```

Wait SAML Did this AGES ago!!!

- The SAML 2.0 spec will let you specify a subject in an Authentication Request
  - But if a subject is specified in the request, the assertion that returns MUST correspond to that subject
  - This is useful for Continuation but not for Bootstrapping
- OpenID Connect offers two hint options:
  - `login_hint` parameter has no return requirement, data is used or ignored at the discretion of the identity provider
  - `id_token_hint` parameter requires a related return, like SAML but far more context is passed

RSAConference2015

San Francisco | April 20-24 | Moscone Center



#RSAC

Quick Demo

How might this tie together?

Context	IDP Pre-Auth	IDP Post-Auth	IDP Pre-Token	IDP at Subsequent Authentication Request
RP Claimed				ID Token hint supplied
User Claimed	Login Hint: pdingle			pdingle (from id_token_hint)
Client Claimed	client id: HR Web App - no secret	client id: HR Web App - no secret	client id: HR Web App - secret provided	client id: HR Web App - no secret
Device Claimed	device id: pam's laptop registered to: pdingle	device id: pam's laptop registered to: pdingle	device id: pam's laptop registered to: pdingle	device id: pam's laptop registered to: pdingle
Network Claimed	network id: 10.10.1.2 -corporate intranet	network id: 10.10.1.2 - corporate intranet	network id: 10.10.1.2 - corporate intranet	network id: 64.20.122.3 - common location for pdingle
IDP Claimed	no IDP session	Session: pdingle	Session: pdingle	Session: pdingle
MFA Claimed	no 2nd Factor	device id: 587 registered to: pdingle	device id: 587 registered to: pdingle	reconfirmed: 587/pdingle

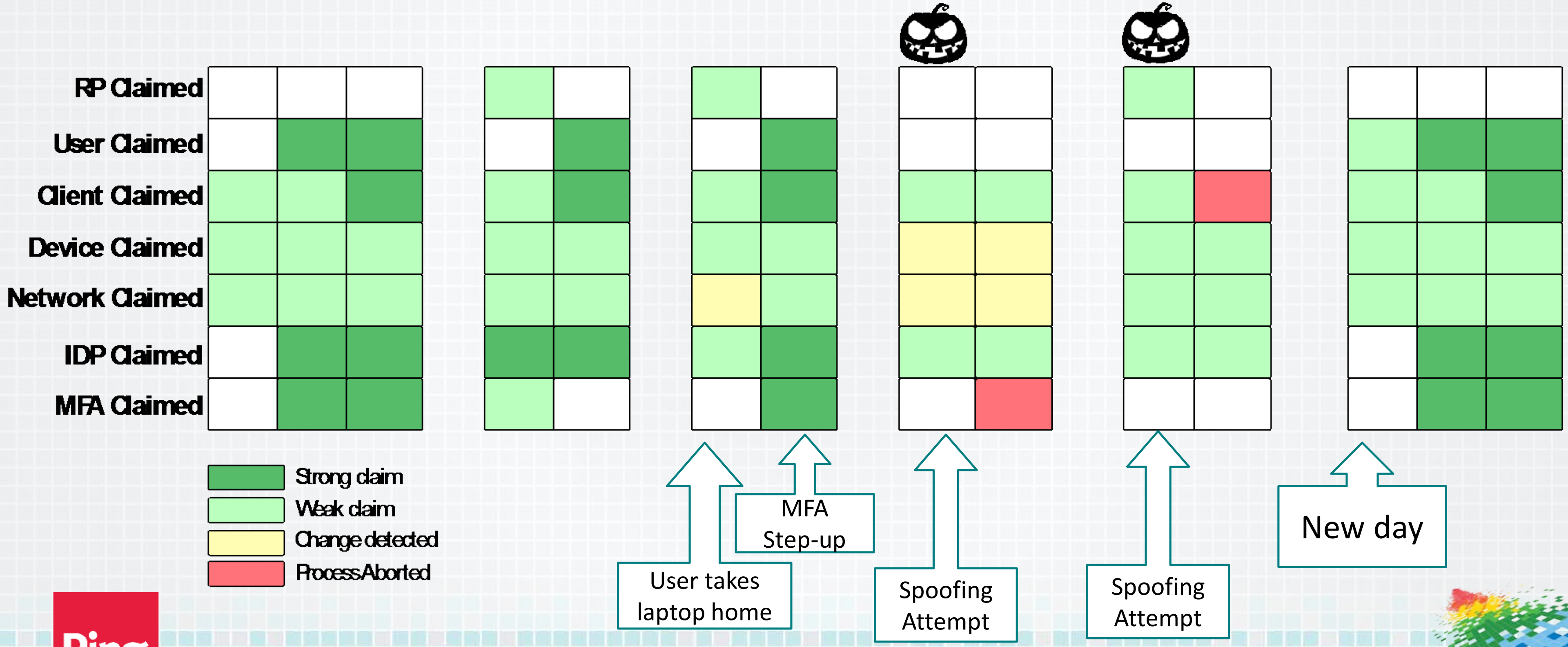
Authentication Ceremony

code sent

id_token sent

The result is a ribbon where anomalies pop

user pdingle & client HRWeb App IDP example intervals over time



RP Claimed				🟢		🟢				🟢				
User Claimed		🟩	🟩		🟩		🟩					🟢	🟩	🟩
Client Claimed	🟢	🟢	🟩	🟢	🟩	🟢	🟩	🟢	🟢	🟢	🟥	🟢	🟢	🟩
Device Claimed	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟨	🟨	🟢	🟢	🟢	🟢	🟢
Network Claimed	🟢	🟢	🟢	🟢	🟢	🟨	🟢	🟨	🟨	🟢	🟢	🟢	🟢	🟢
IDP Claimed		🟩	🟩	🟩	🟩	🟢	🟩	🟢	🟢	🟢	🟢		🟩	🟩
MFA Claimed		🟩	🟩	🟢			🟩		🟥				🟩	🟩

- 🟩 Strong claim
- 🟢 Weak claim
- 🟨 Change detected
- 🟥 Process Aborted

User takes laptop home

MFA Step-up

Spoofing Attempt

Spoofing Attempt

New day

What would this look like in Enterprise Identity Architectures?

- Identity Providers
  - Accept Login Hints in federated authentication requests
    - Start by simply populating the login form
  - Accept id_token hints
    - Consider them login hints to start
  - Log that context, start looking for patterns
- Relying Parties
  - Call Account Chooser as part of IDP discovery routine and place login hints in the authentication request
    - See http://openid.net/ac for details
  - Work with identity providers on caching id_tokens and providing them as hints for session renewal
  - Take a good look at context switching use cases – most common in consumer RPs but have an application around administrator use cases too

https://www.flickr.com/photos/hugo90/4455412652

Future of Federated Context Sharing

- Shared Signals/ATOC
  - Goal is to prevent cascading identity fraud on the internet by sharing significant identity events for use as context in other domains
    - Moving into a working group at the OpenID Foundation
- Device Posture
  - Use case is strong to send this information in both directions
  - Most SaaS apps are unable to alter user experience on a session-by-session basis

Apply What You Have Learned Today

- Enterprises
  - Examine your Authentication Ceremony
    - Simple start: try deploying account chooser at the IDP
    - Look at whether your SaaS apps support a subject in the SAML AuthnRequest
- Apps: Examine your IDP Discovery
  - Are you asking for user identifiers and discarding the user information?
  - Consider adding that data to the SAML authentication request
  - If you already use OAuth or OpenID Connect, play with login hints
- Everyone: Just start collecting
  - If you collect now, then when you are ready, you have a body of historical data to tune your systems with

Conclusion

- When treated as additional context to an authentication, context sent from relying parties can improve usability and add useful data to adaptive security evaluations.
- Little was available to identity architects in the areas of bootstrapping, continuation, and context switching until now, but options are opening up
- id_token_hints can enable extremely in-depth tracking of every authentication request/response
- Consistent use of choosers and login hints can create a “ceremony” both at the machine and the user level that provides cues to abuse

Further Reading/Information

AccountChooser WG: http://openid.net/ac

AccountChooser example: http://hipstabank.com

Google Identity Toolkit:
https://developers.google.com/identity-toolkit/

Web: http://pingidentity.com

Twitter:

- @pingidentity
- @pamelarosiedee

https://www.flickr.com/photos/gideonvanderstelt/3833757689

* Gratuitous kitten picture included for express purpose of annoying @paulmadsen