Componentization of Security Principles

Justin Richer
The MITRE Corporation
202 Burlington Road
Bedford, MA 01730
jricher@mitre.org

Dazza Greenwood
MIT Media Lab
1 Broadway, 14th Floor
Cambridge, MA 02142
dazza@civics.com

Bruce Bakis
The MITRE Corporation
202 Burlington Road
Bedford, MA 01730
bbakis@mitre.org

ABSTRACT
In this paper, we present a position on the value of componentizing security principles. We then discuss a set of emerging technologies that are able to make full use of such componentized principles. Finally, we present a high level case study of one company's deployment of this technology.

1. INTRODUCTION
Security of computer systems is not a simple topic. As desirable as it would be, there exists no control that could be turned to "Maximum Security" without affecting other factors such as cost or usability. In reality, all security decisions are made in a complex context with various tradeoffs.

2. COMPONENTIZATION OF SECURITY
Recognizing security as a contextual problem offers an occasion to view different aspects of security in a componentized way. These components depend on the context in which the security decision is being made, and some components follow here.

2.1 Control of Credentials
It is not uncommon for an application developer to wish to be in control of the entire user experience, including all matters of security such as a user's credentials. However, if many different applications take this view, end users will need to manage many different sets of credentials. This creates both a usability problem as well as a security problem, as the coping mechanisms that people use with this situation include re-using passwords with different systems and using weak but memorable passwords.

As an alternative to this, an application can authenticate users through a federated identity protocol. This method allows a better user experience, as users are able to re-use an identity. At the same time, it increases the security posture of the entire network by decreasing the proliferation of passwords and weak credentials.

2.2 Credential Binding
Many classifications of federated identity systems conflate different aspects of identity, such as identity proofing, credential presentation, and verifiability of an assertion, into a single categorization, such as the NIST Level of Assurance [1]. While such shorthand is sometimes useful, we believe that there is value in separating the different aspects of identity federation and considering them orthogonally.

Copyright is held by the author/owner. Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee.

Symposium on Usable Privacy and Security (SOUPS) 2014, July 9-11, 2014, Menlo Park, CA.

For instance, take an identity provider that is capable of providing a very high assurance credential to the user, such as multi-factor cryptographic authentication, and federates that identity through a strong and verifiable protocol but does no identity proofing of the individual in question. A user wants to access sensitive personal information with such an identity. By decomposing the different aspects of their digital identity from one another, we can see that they are presenting a very high level *credential* with a very low-level identity *binding*. Here, the identity binding between a credential and real person can happen out of band. For instance, a patient sitting in their doctor's office can present their insurance card and driver's license at the same time they log in with their identity provider. By doing a late binding, we believe we can better leverage digital identities.

2.3 Authentication and Authorization
Traditional application design places authentication at the root of all authorization decisions: if the system can figure out who the user is, it can make the authorization decision. However, an application needs to answer one fundamental security question: should the action that is being requested be allowed, or not? We believe that the authentication of the party requesting the action is one of many aspects that need to be considered. By separating the authentication of the user from the authorization of the action, one can increase the security posture of the application. Many internet services do heuristic processing of requests from authenticated users to detect fraudulent account activity. Additionally, the rise of web application programming interfaces (APIs) and mobile applications has driven the adoption of authorization delegation.

3. TECHNOLOGY COMPONENTS
Componentized security can be found on the public internet, where the need to work across traditional security domain boundaries has driven development of a set of technologies to support this. In particular, OAuth 2.0 [2] provides an authorization and delegation mechanism used to protect a wide variety of APIs, and OpenID Connect [3] builds on top of OAuth 2.0 to define a distributed identity protocol. These and related technologies enable the deployment of the security concepts discussed herein.

3.1 Dynamic Discovery and Onboarding
Traditionally, security architectures assume that all parties are known ahead of time. However, many use cases require an amount of flexibility that is not possible with a static approach. We contend that a technology must be dynamic enough to account for such an environment.

Instead, a system architecture that enables the simple discovery of services through minimal inputs and information can facilitate the wide and varied use of APIs and services. In this environment, new client applications need to be onboarded easily and

automatically. OpenID Connect enables this through a dynamic discovery of the services and dynamic registration of the clients.

3.2 Runtime Security Decisions

In a dynamic environment where the components are able to connect to each other without prior knowledge, there needs to be a mechanism for an appropriate party (such as an end user) to approve or deny a given authorization request as that request is made. We believe that this "Trust On First Use" (or TOFU) model is a powerful mechanism that allows security components to scale in a controlled and trustable fashion. OpenID Connect uses this pattern to provide user authentication across security domains.

3.3 Federated Identity as a Technological Abstraction Layer

Within a security domain, federation technologies can provide an abstraction layer to client applications by using well-defined open standards. The use of a standard protocol increases the availability of libraries and tools to the developer. We believe that this can increase developer acceptance and adoption, leading to decreased reliance on custom-built and little-understood security code.

3.4 Managing Differential Deployment

By acting as an abstraction from underlying systems, federation can help facilitate differential deployment of new technologies. Old federation endpoints can be run in parallel with new endpoints, with the same identity stores, accommodating both legacy and novel systems on relatively equal footing. Clients can transition off of the legacy protocol gradually without requiring a mass migration and providing instead an elegant evolution path.

4. CASE STUDY

The MITRE Corporation has deployed a federated identity service that embraces a componentized security architecture model and makes use of federated identity technologies.

4.1 MITREid

In 2009, the MITRE Corporation deployed the MITREid OpenID service based on three core tenets: use of existing identities, runtime trust decisions, and open standard protocols.

MITREid is coupled to the corporate infrastructure, giving all current employees at MITRE digital identities through the service. These identities can be used with any OpenID 2.0 compliant website, both inside and outside the company firewall. To date, over 8000 users have logged in to more than 400 different sites using the OpenID 2.0 service. While a small number of company-provided services are whitelisted by the server, the vast majority of logins with this system are driven by end-user decisions made at runtime using a TOFU model.

The service was developed around the well-established OpenID 2.0 authentication standard [4], giving developers access to many existing libraries. One internal prototype deployed prior to the MITREid service had built-in support for OpenID and worked immediately, as did compliant external sites.

4.2 MITREid Connect

In 2012, MITREid was expanded with support for the newer OpenID Connect protocol in MITREid Connect, deployed on the same infrastructure and with the same core tenets as the original MITREid. The OpenID Connect standard was finalized in February 2014, and while the development community around the protocol is still young, adoption is growing rapidly. The legacy OpenID 2.0 service remains available in parallel to the OpenID Connect deployment, and new applications are encouraged to make use of OpenID Connect where possible.

The MITREid Connect implementation [5] is open source, and in the fall of 2013, the MITRE Corporation transferred stewardship of the MITREid Connect open source project to the MIT Kerberos and Internet Trust Consortium (KIT) [6]. Both MITRE and KIT hold co-copyright and have committed to keeping the project open source and available under the Apache 2.0 license.

4.3 Certification Through Kantara

In 2013, MITREid was accredited through the Kantara Initiative [7]. We found that our approach of providing digital identities to all current employees was novel. We approached digital identity as an IT service much the same as email or telephone service while the accreditation process today assumes identity providers offer identity as a service to customers.

4.4 Challenges and Lessons Learned

A challenge faced by MITREid has been overcoming the inertia of existing practices and systems. OpenID was made for the public internet, and such a decentralized model was not typical in the enterprise. To mitigate this, MITREid is an extension of the existing identity infrastructure, allowing people to re-use existing identity profiles and credentials in a new way. Developers can access this existing population of users with relative ease.

Another challenge has been that business policy and legal contracts are not well suited for the type of dynamic environment that this technology enables. Methods for addressing this disparity are an active area of research today.

5. CONCLUSION

In conclusion, we propose that the adoption of a componentized security model be built on federated identity technologies and open standards. We contend that enterprises should provide their users with portable digital identities just like email or telephone. Even though there are still obstacles to overcome, the benefits enabled by this approach make it worth investigating.

6. REFERENCES

[1] Burr, W., et al, "Electronic Authentication Guideline," NIST Special Publication 800-63-2, August 2013, http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-63-2.pdf

[2] Hardt, D., "The OAuth 2.0 Authorization Framework," IETF, RFC6749, http://tools.ietf.org/html/rfc6749

[3] Sakimura, N., et al, "OpenID Connect Core 1.0," OpenID Foundation, February 2014, http://openid.bitbucket.org/openid-connect-core-1_0.html

[4] The OpenID Foundation, "OpenID Authentication 2.0", http://openid.net/specs/openid-authentication-2_0.html

[5] MITREid Connect, https://github.com/mitreid-connect

[6] MIT Kerberos and Internet Trust, http://kit.mit.edu/

[7] Kantara Initiative, http://kantarainitiative.org/

© 2014 The MITRE Corporation. All rights reserved.
Approved for Public Release: Distribution Unlimited (Case #14-1647)