Building a Secure PI Web API Environment

Presented by **Mike Sloves**
Ray Verhoeff

User Conference 2017 Themes

What do we mean by secure?

- Basic summary of security concepts:
  - Minimizing the “Attack Vector”
  - Preventing various attacks
    - Man in the Middle
    - DDoS
    - Etc.
  - *Staying inside your firewall does not make you immune*
- What we are doing to help secure PI Web API?

What is PI Web API? (briefly…)

- RESTful Service
- Any client platform, language, etc.
- Modern method of supporting any device
- Away from your site using mobile

Methods of Securing PI Web API

- PI System Security
- Certificates
- Authentication
- Cross-Origin Resource Sharing (CORS)
- Cross-Site Request Forgery (CSRF)
- (Distributed) Denial of Service
- IT Resources

What is a Certificate?

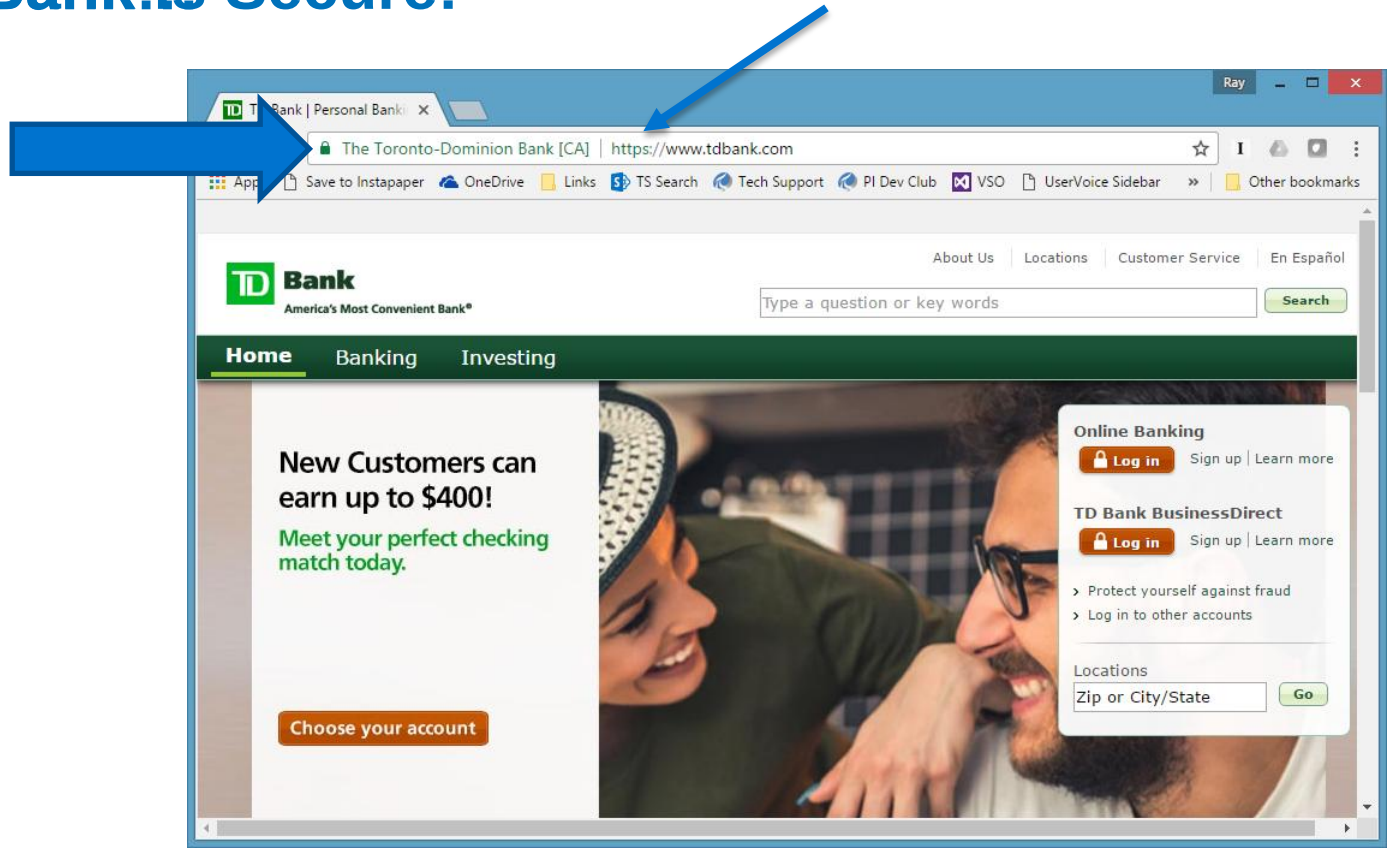
An electronic document used to prove the ownership of a public key

- Information about the key
- Information about its owner's identity
- Digital Signature of a verifying entity

Does this all check out?

YAY!!!!

My Bank.is Secure!



Certificates

- How encryption works
- PI Web API has no HTTP option
  - Why we insist on certificates
  - *“But I’m just doing development!”*
- Getting a certificate is not difficult

How Certificates are used

- Client and Server negotiate:
  - SSL/TLS version
  - Ciphersuite
  - Compression (if any)
- Client confirms that the Server's certificate is valid
- Client and Server exchange keys to use for encryption

How to Get a Certificate

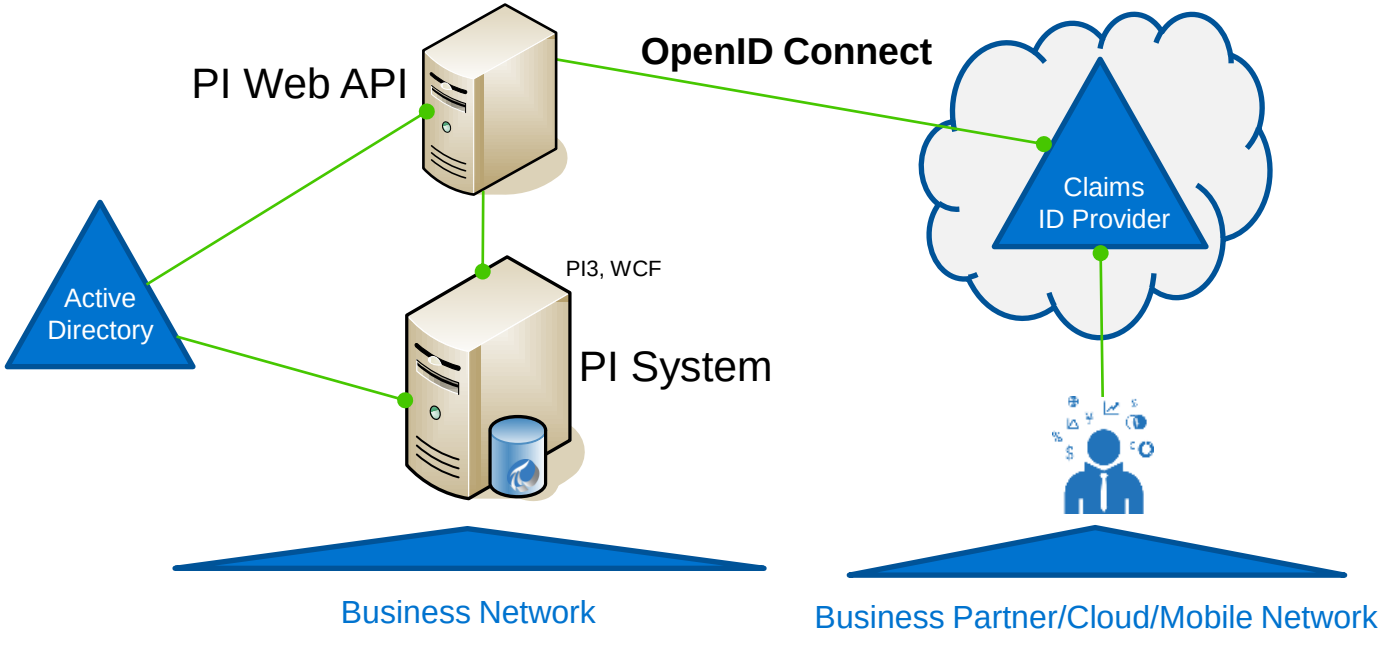
- Self-signed
- Buy one from a certificate vendor
  - Verisign
  - Geotrust
  - Comodo
  - Digicert
  - *Lots of others…Look it up, we did…*
- Letsencrypt.com
  - Free certificates! Becoming more popular
- Windows Domain Certificate

Authentication

- Anonymous
- Basic
- Kerberos
- *And introducing…*

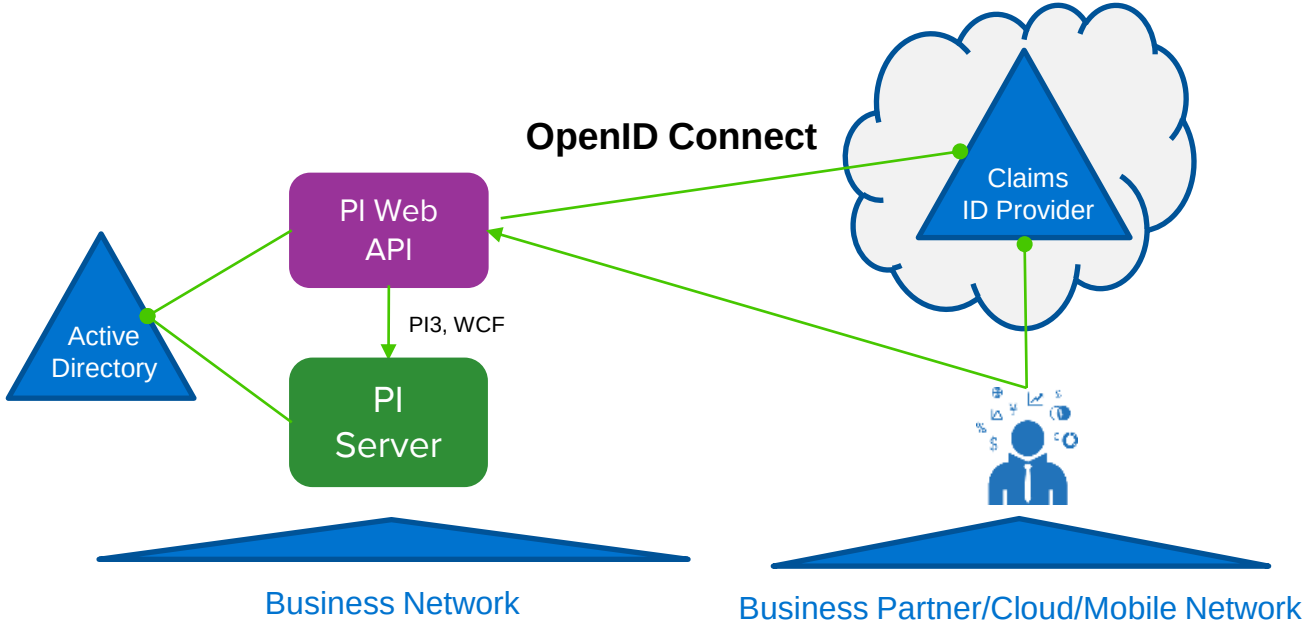
Claims-Based Authentication

- Login using an external Identity Provider
  - *No need to expose corporate AD credentials*



Claims-Based Authentication

- Login using an external Identity Provider
  - *No need to expose corporate AD credentials*



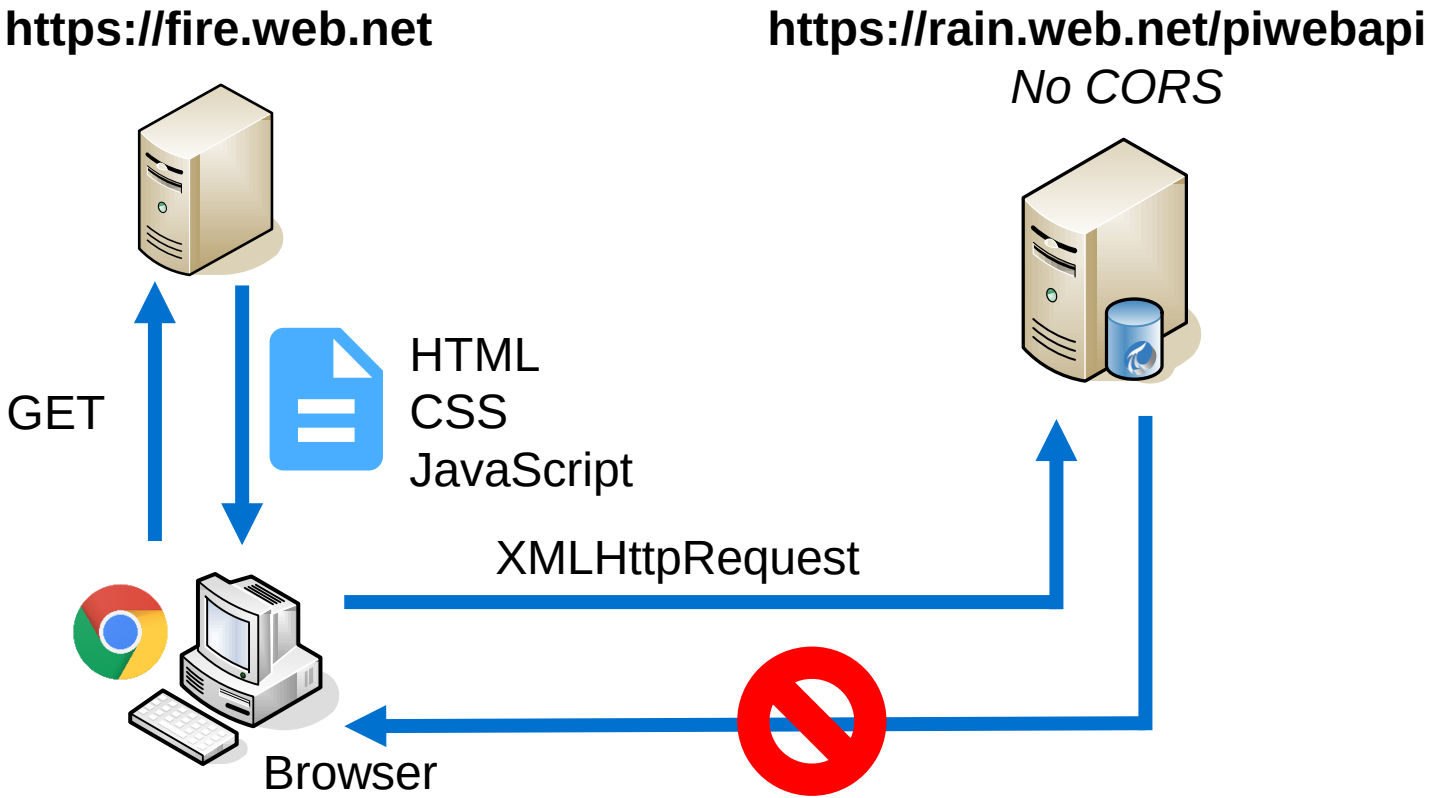
How we did it

- OpenID Connect
  - An Authentication layer on top of OAuth 2.0
  - Controlled by the OpenID Foundation
  - RESTful HTTP API using JSON format
  - Wide acceptance in the industry

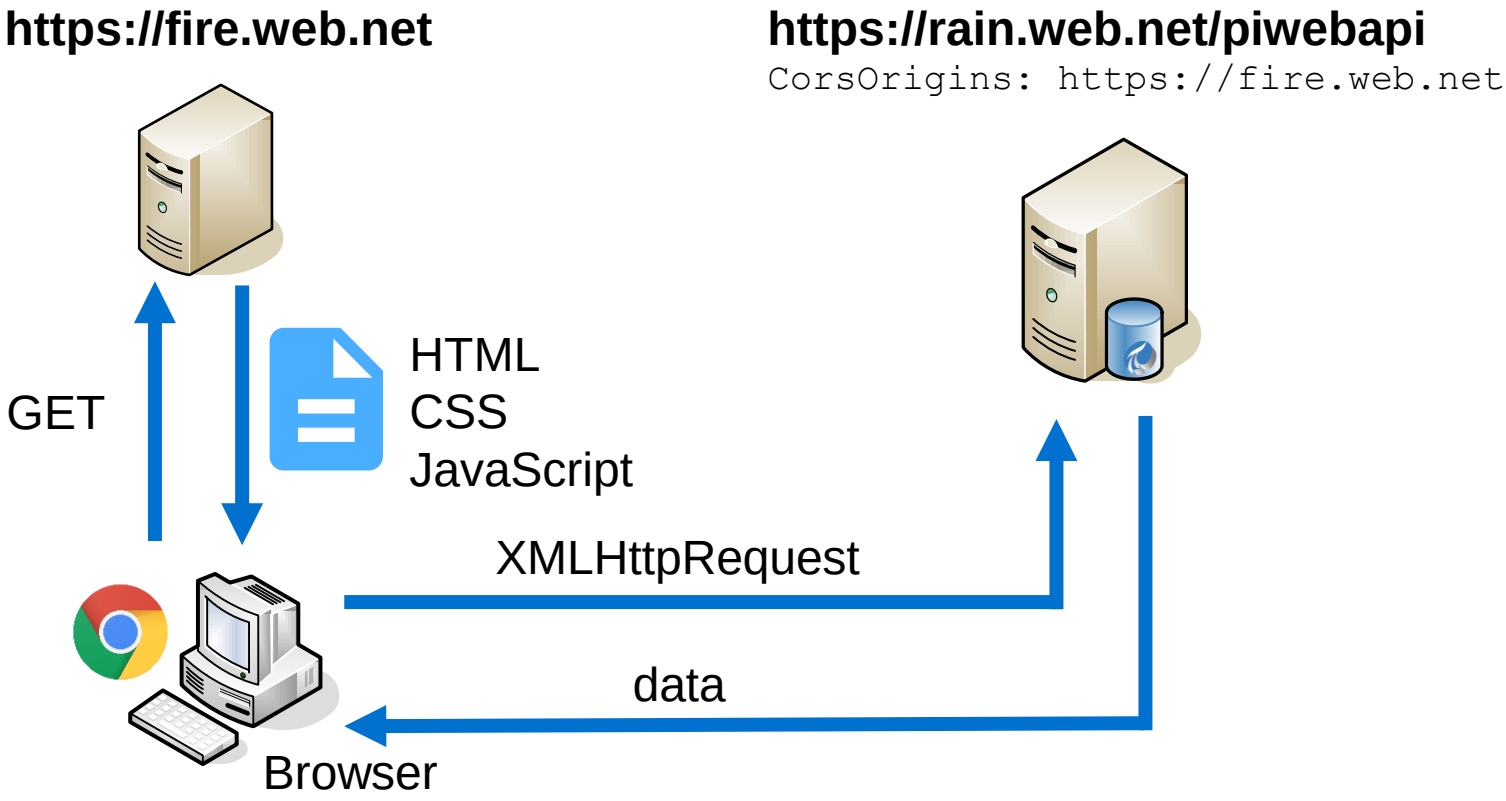
Response from /.well-known/openid-configuration


```
{
  "authorization_endpoint":"https://login.windows.net/59edc9e0-ed80-436b-b179-554c9b5eff79/oauth2/authorize",
  "token_endpoint":"https://login.windows.net/59edc9e0-ed80-436b-b179-554c9b5eff79/oauth2/token",
  "token_endpoint_auth_methods_supported":[
    "client_secret_post",
    "private_key_jwt"
  ],
  "jwks_uri":"https://login.windows.net/common/discovery/keys",
"id_token_signing_alg_values_supported":[
    "RS256"
  ],
  "http_logout_supported":true,
  "frontchannel_logout_supported":true,
  "end_session_endpoint":"https://login.windows.net/59edc9e0-ed80-436b-b179-554c9b5eff79/oauth2/logout",
  "response_types_supported":[
    "code",
    "id_token",
    "code id_token",
    "token id_token",
    "token"
  ],
  "scopes_supported":[
    "openid"
  ],
  "issuer":"https://sts.windows.net/59edc9e0-ed80-436b-b179-554c9b5eff79/",
  "claims_supported":[
    "sub",
    "iss",
    "cloud_instance_name",
    "aud",
    "exp",
    "iat",
    "auth_time",
    "acr",
    "amr",
    "nonce",
    "email",
    "given_name",
    "family_name",
    "nickname"
  ],
}
```

Cross-Origin Resource Sharing (CORS)



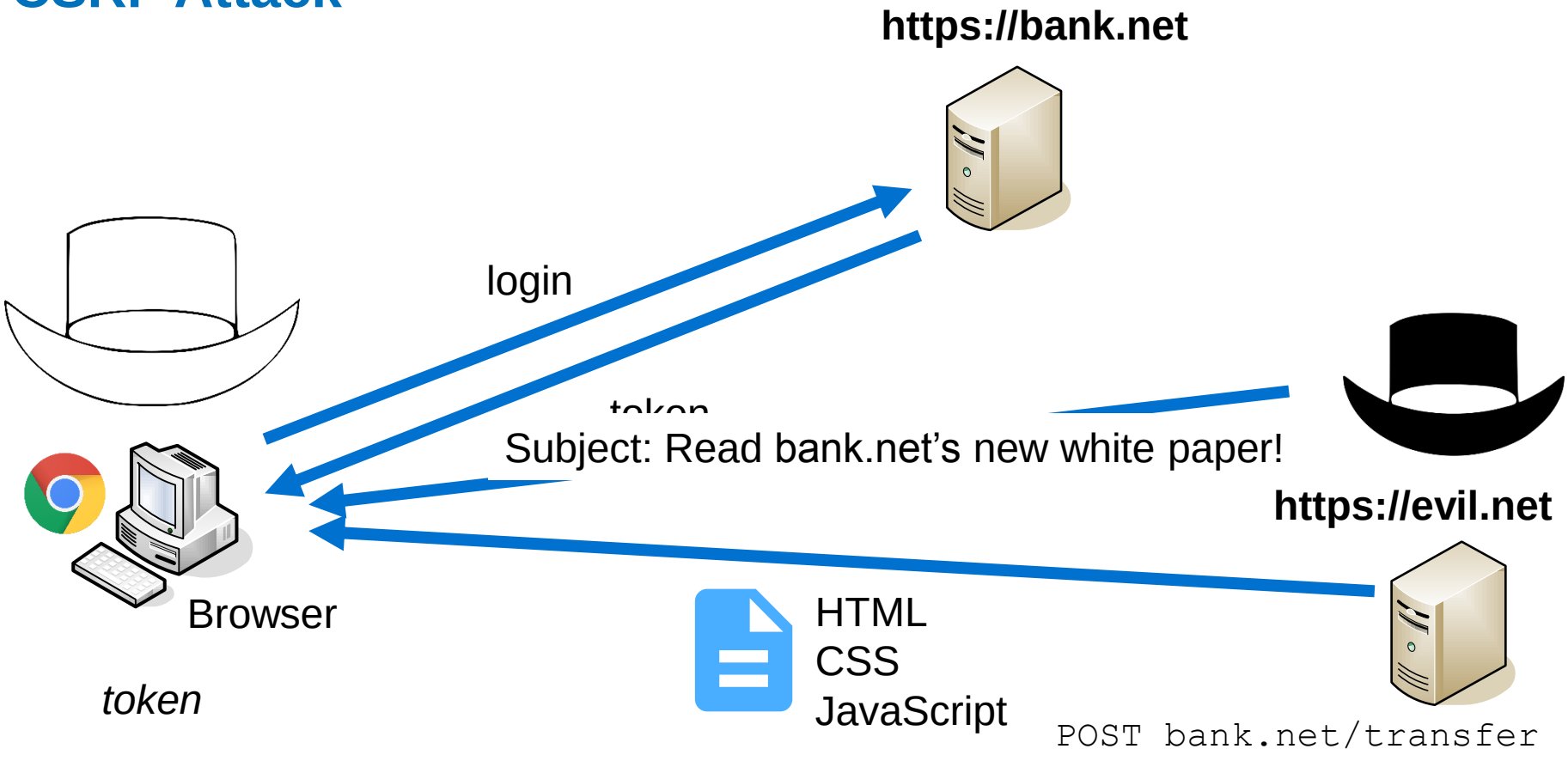
Cross-Origin Resource Sharing (CORS)



Cross-Site Request Forgery (CSRF)

- You log into a legitimate website
  - *Your browser keeps the authentication token*
- You are tricked into visiting a bad website
  - *JavaScript containing evil code downloaded*
- JavaScript executes
  - *Your authentication token is used*
  - ***Evil code does damage to your legitimate website!***

CSRF Attack



Cross-Site Request Forgery (CSRF)

- Existing applications may need to be updated!
  - Good News: GET is fine
  - Bad News: POST, PATCH, DELETE need attention
    - Add Header:

      ```
      X-Requested-With: XMLHttpRequest
      ```

    - Good News (after the Bad)
      - Modern browsers do this for you

Custom Headers

- HTTP defines a long list of request and response headers
- Some instruct browser on which rules to enforce and how
- *Example:*
  - `Referer: strict-origin`

Custom Headers

- PI Web API 2017 R2 introduces **custom headers**
- You can create your own custom headers with values
- Your settings override ours
- *Don’t use this to disable security settings!*
- Example:
  - `Content-Security-Policy: unsafe-eval`

Online Security Audit Tools

- https://observatory.mozilla.org/
- https://securityheaders.io/

(Distributed) Denial of Service

- Throughput limits implemented in PI Web API configuration:
  - `RateLimitMaxRequests`
  - `RateLimitDuration`
  - `MaxReturnedItemsPerCall`
- Set PI Web API to read-only:
  - `DisableWrites` configuration item

YOUR IT Department (not ours…)

- VPN
- Secure Communications
- Disable Writes
  - POST restriction in PI Web API configuration
- Use Load Balancers/Routers/Switches to limit connectivity

Which method should I use?

Checklist

- Intranet
  - ☑ Certificate
  - ☑ CORS
  - ☑ CSRF
  - ☑ Authentication Model
  - ☑ Target Platforms
  - ☑ Denial of Service defenses

Checklist

- Extranet
  - Everything in the Intranet Checklist, plus:
  - Authenticate: Basic or Claims-Based
    - Either way: a *local* Windows Domain Controller is needed to support your user community

Have an idea how to improve our products?

OSIsoft wants to hear from you!

https://feedback.osisoft.com/

감사합니다

Danke

谢谢

Merci

Gracias

Thank You

ありがとう

Спасибо

Obrigado

Questions

Please wait for the **microphone** before asking your questions

State your **name & company**

Please remember to…

Complete the Online Survey for this session

http://bit.ly/uc2017-app

Contact Information

Mike Sloves
msloves@osisoft.com
Group Leader
OSIsoft, LLC

Ray Verhoeff
ray@osisoft.com
Product Manager
OSIsoft, LLC