
TRENDS IN WEB VULNERABILITIES
MICHEL CHAMBERLAND
Trustwave®
SpiderLabs®

Introduction

Agenda

- Introduction
  - Session **Goals**
  - **Presenter** and **Trustwave SpiderLabs** background
- Analysis Overview
  - Data Source
  - Most **frequently found SEVERE** vulnerabilities
  - Most **frequently found OVERALL** vulnerabilities
- About the Vulnerabilities
  - **What** are they?
  - How to **fix** them
  - **Why** you should **care**
- Conclusion

Trustwave® SpiderLabs®

Introduction

Session Goals

- Provide an overall sense of the state of web application vulnerabilities that are **commonly found** by professional penetration testers
- Provide you with the tools to **identify overlooked areas** in your web applications.
- Help you determine where to **focus your efforts** in order to help your organization

Trustwave® SpiderLabs®

Introduction

About the Presenter

- **Michel "Mike" Chamberland**, MSc
- **North America Practice Lead** with Trustwave SpiderLabs
- CISSP, OSCP, OSWP. CEH, CHFI, CCSK, MCP, GIAC G2700, MCTS, Security+, etc.
- Grew up in **Sherbrooke, Qc** and now lives in Sarasota, FL
- Works closely with all **USA and Canada** based SpiderLabs resources

Trustwave® SpiderLabs®

Introduction

About Trustwave SpiderLabs

- A **division within Trustwave**
- Consists of **150+ specialized security experts**
- Focuses on **penetration testing, research and incident response**
- Performed **millions of scans** and **thousands of penetration tests**
- Over **9 million web application attacks** researched last year
- **97%** of applications tested by Trustwave **had one or more vulnerability**

Trustwave® SpiderLabs®

Analysis Overview

Trustwave® SpiderLabs®

Analysis Overview

Data Source

- Based on an **analysis of vulnerabilities found** by the Trustwave SpiderLabs team over the last few years
- Will cover both **most frequently found overall** as well as **most frequently found severe** vulnerabilities.
- Vulnerabilities will be **grouped based on similarities** and then discussed in further details
- **Statistics on vulnerabilities and vulnerability groups** will be explored

Trustwave® SpiderLabs®

Analysis Overview

Data Source

- Vulnerability **data collected from thousands of web application security assessments** performed
- All **data collected is anonymous** and not associated with any specific organizations
- Trustwave serves over **3 million customers in 96 countries**
- The customer base is **spread across all verticals**
- Many of these customers are in the **educational sector**

Trustwave® SpiderLabs®

What is a Severe Finding

Types of Severe Findings

- What is **defined as severe**:
  - Critical
  - High
- Combined, they make up approximately **10% of findings**

Trustwave® SpiderLabs®

What is a Severe Finding

Critical Severity Findings

- The **attack scenario** tested in this exercise **succeeded, and resulted in a systems compromise**
- Exploitation is **trivial**
- Exploitation requires **no authentication**, or authentication is available to a **member of the public with minimal effort**
- Exploitation results in a **large-scale loss of sensitive information or tangible assets**
- A strong **need for immediate corrective measures** exists

Trustwave® SpiderLabs®

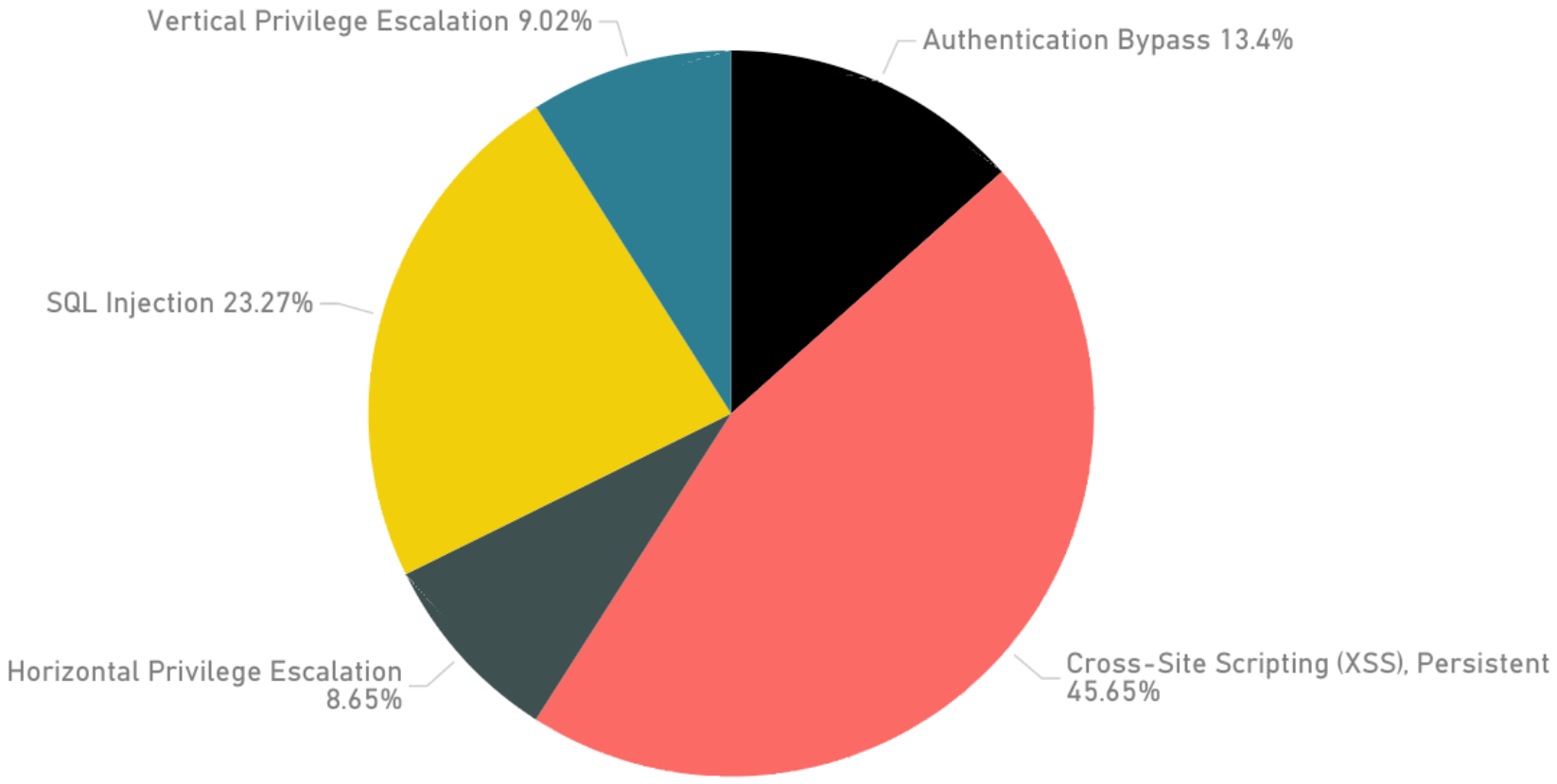
What is a Severe Finding

High Severity Findings

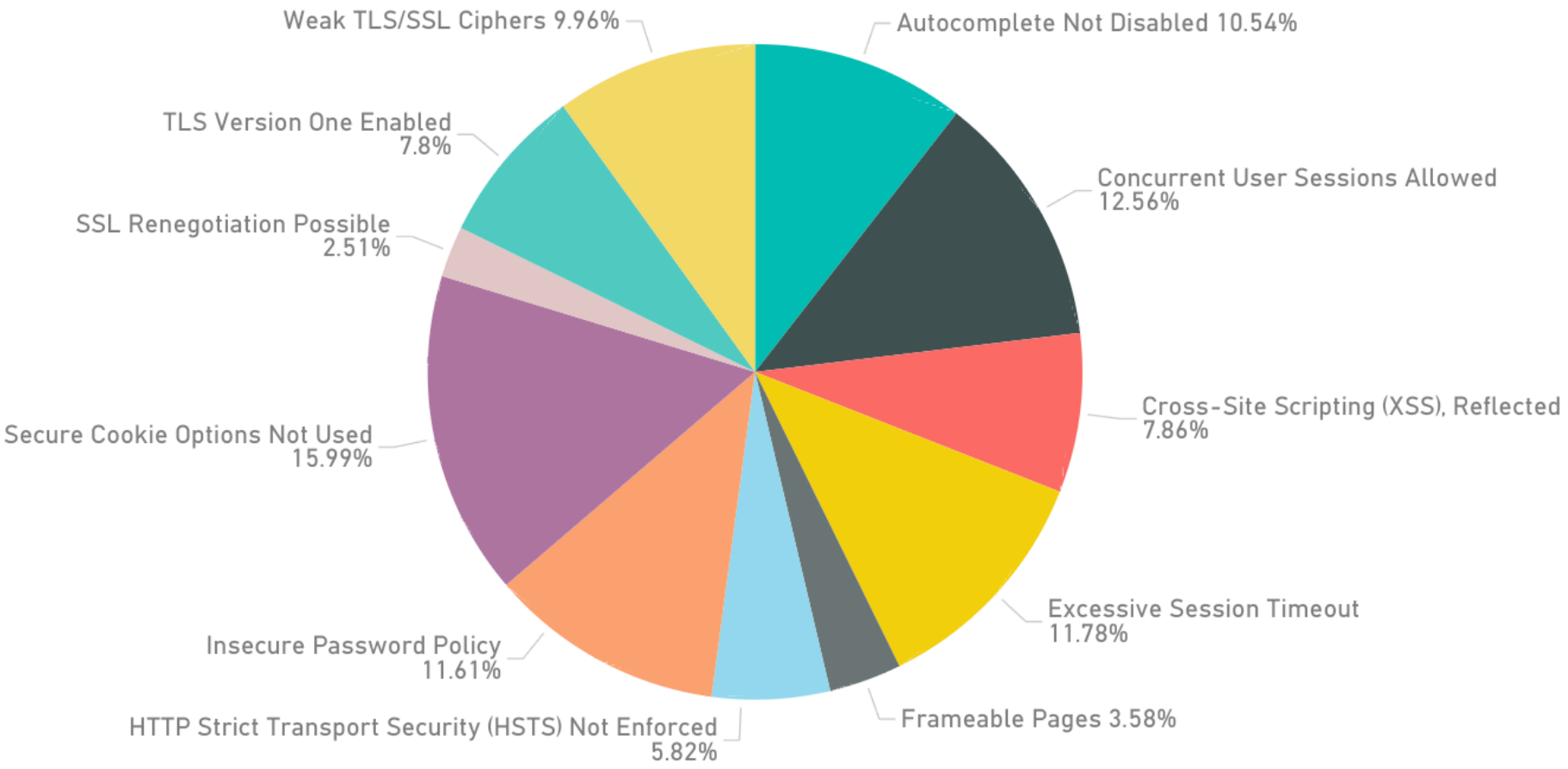
- The **attack scenario** tested in this exercise **succeeded, and resulted in a systems compromise**
- The attack can only be performed by an **authenticated user**
- Technical vulnerability details and/or exploit code are **publicly available**
- An **additional attack vector may be needed** to craft a successful attack using this exploit, but that **vector is trivial**
- Exploitation of the vulnerability (1) may result in the **costly loss of sensitive data or tangible assets**, or (2) may significantly **violate, harm, or impede the organization's mission, reputation, or interest**.
- A strong **need for corrective measures** exists

Trustwave® SpiderLabs®

Most Frequent Severe Findings



Most Frequently Found Overall



The Vulnerabilities

Trustwave® SpiderLabs®

The Vulnerabilities

- The Rock Stars
- The Heavy Hitters
- Forgotten Killers
- The Misunderstood
- The Personal Problems

Trustwave® SpiderLabs®

The Rock Stars

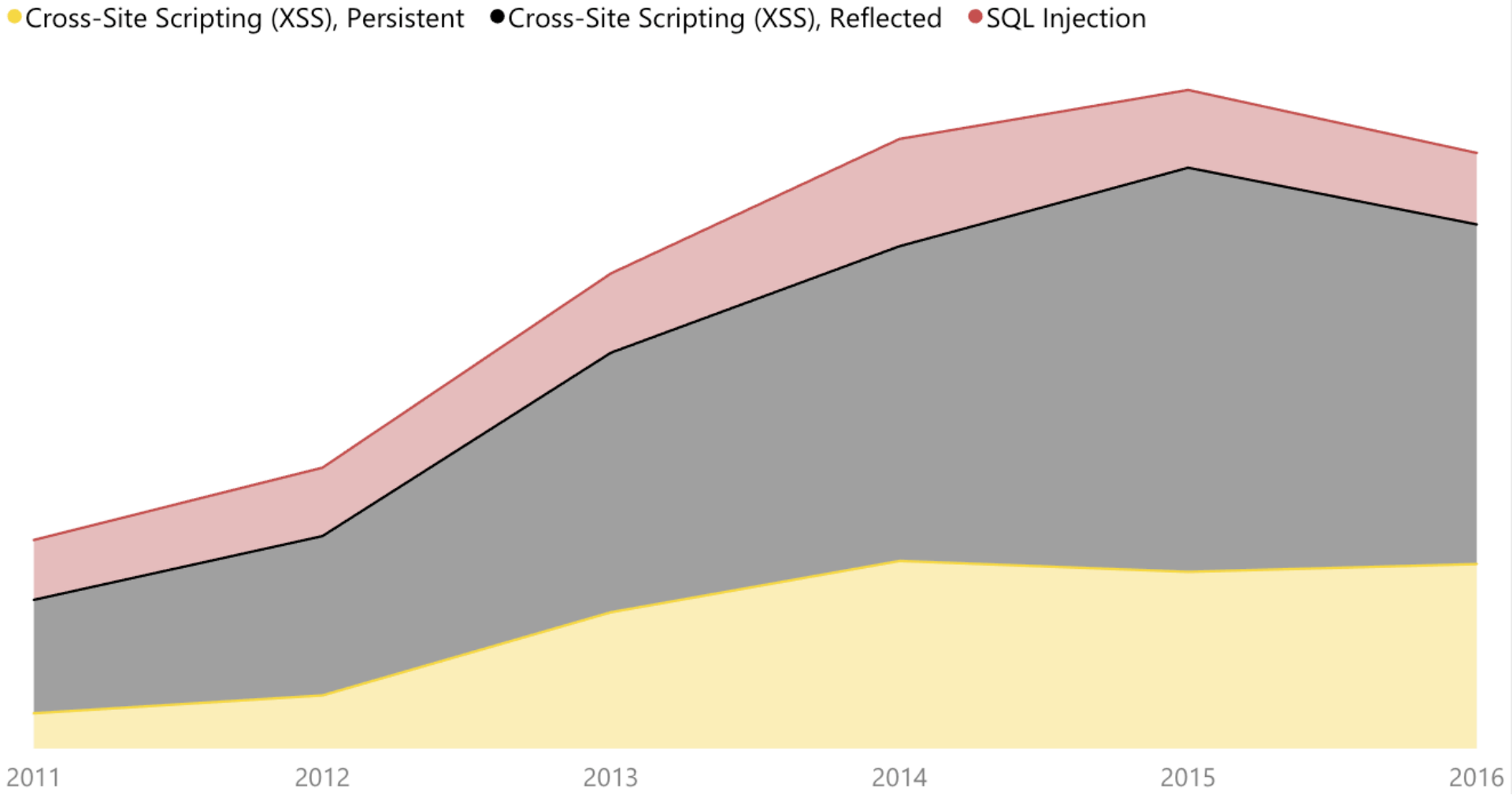
The Rock Stars

Description

- These are the vulnerabilities **everyone knows about** and almost always result in a **severe impact**
- Used to be found **systemically**
- **SQL Injection**
  - Allows an attacker to insert arbitrary commands into a database query or statement.
- **Cross-site Scripting**
  - Occurs when web applications do not properly validate user-supplied inputs before including them in dynamic web pages.

Trustwave® SpiderLabs®

The Rock Stars



The Rock Stars

Why they matter

- Cross-Site Scripting
  - **Session hijacking**
  - Can be used to **virtually deface** web applications
  - **Social engineering** (login prompts, fake updates, payment form, etc.)
  - **Redirect** users to another site
  - **Tunneling and network discovery**
  - Log user's **keystrokes**
- SQL Injection
  - **Exfiltration or tampering** of data
  - Get **operating system level access**
  - **Escalate** privileges

Trustwave® SpiderLabs®

The Rock Stars

Mitigation

- Define a solid application **architecture**
  - Don't fix each instances individually
- **Sanitize user input**
- Prefer a **white listing** approach
- Use **prepared statements and stored procedures** for all SQL operations
- Validate input on **both client and server side**
- **Escape output** before storing in database or rendering in web browser

Trustwave® SpiderLabs®

The Heavy Hitters

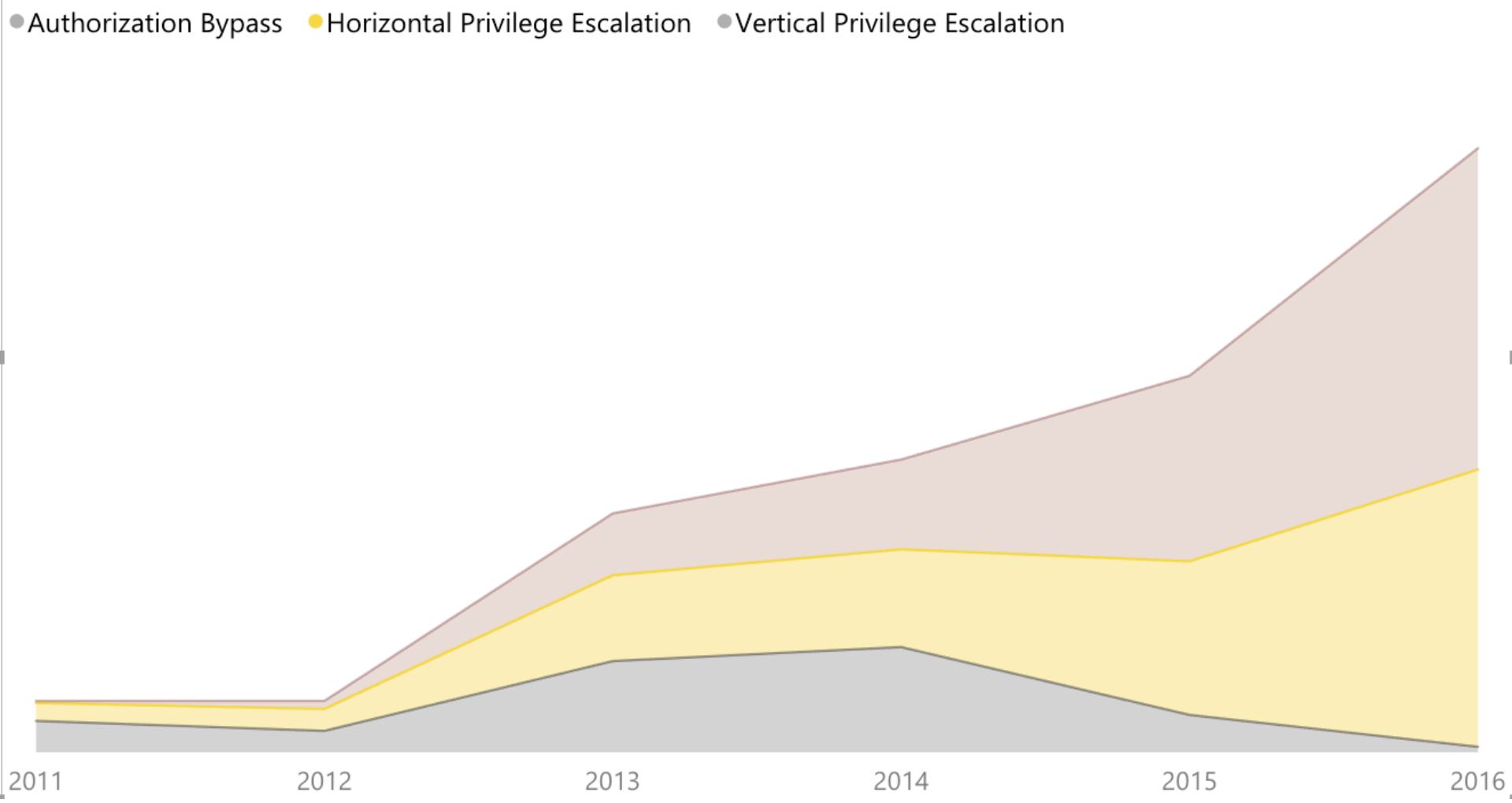
The Heavy Hitters

Description

- Caused by **poor authentication and authorization controls**
- These are vulnerabilities that are **often overlooked** but almost always result in a **severe impact**
- Today's **tools do not do a good job** at finding these types of vulnerabilities
- Authentication Bypass
  - **Valid session** identifier is **not required** to access resources
- Vertical Privilege Escalation
  - Authorization controls are not properly enforced, allowing **unauthorized access to resources or functions**
- Horizontal Privilege Escalation
  - Ability to **view, delete or modify other user's data**

Trustwave® SpiderLabs®

The Heavy Hitters



The Heavy Hitters

Why they matter

- Almost always lead to a **severe impact**
- **Overlooked** by **automated tools**
- Often **overlooked** by traditional **application testing**
  - Focused on UI
- Your **firewall, IDS, IPS and/or WAF will not help you**
- Requires **little technical skill** to exploit

Trustwave® SpiderLabs®

The Heavy Hitters

Mitigation

- Fixed at the **architecture level**
- Difficult to successfully implement from scratch
  - **Leverage existing frameworks**
- These authentication and authorization frameworks should:
  - Ensure a **proper session identifier** is associated with each request
  - Ensure the **user's role and permissions** allow the requested action
- Ensure **roles and permissions are defined** and tested
- Review **release management/deployment procedures**

Trustwave® SpiderLabs®

The Forgotten Killers

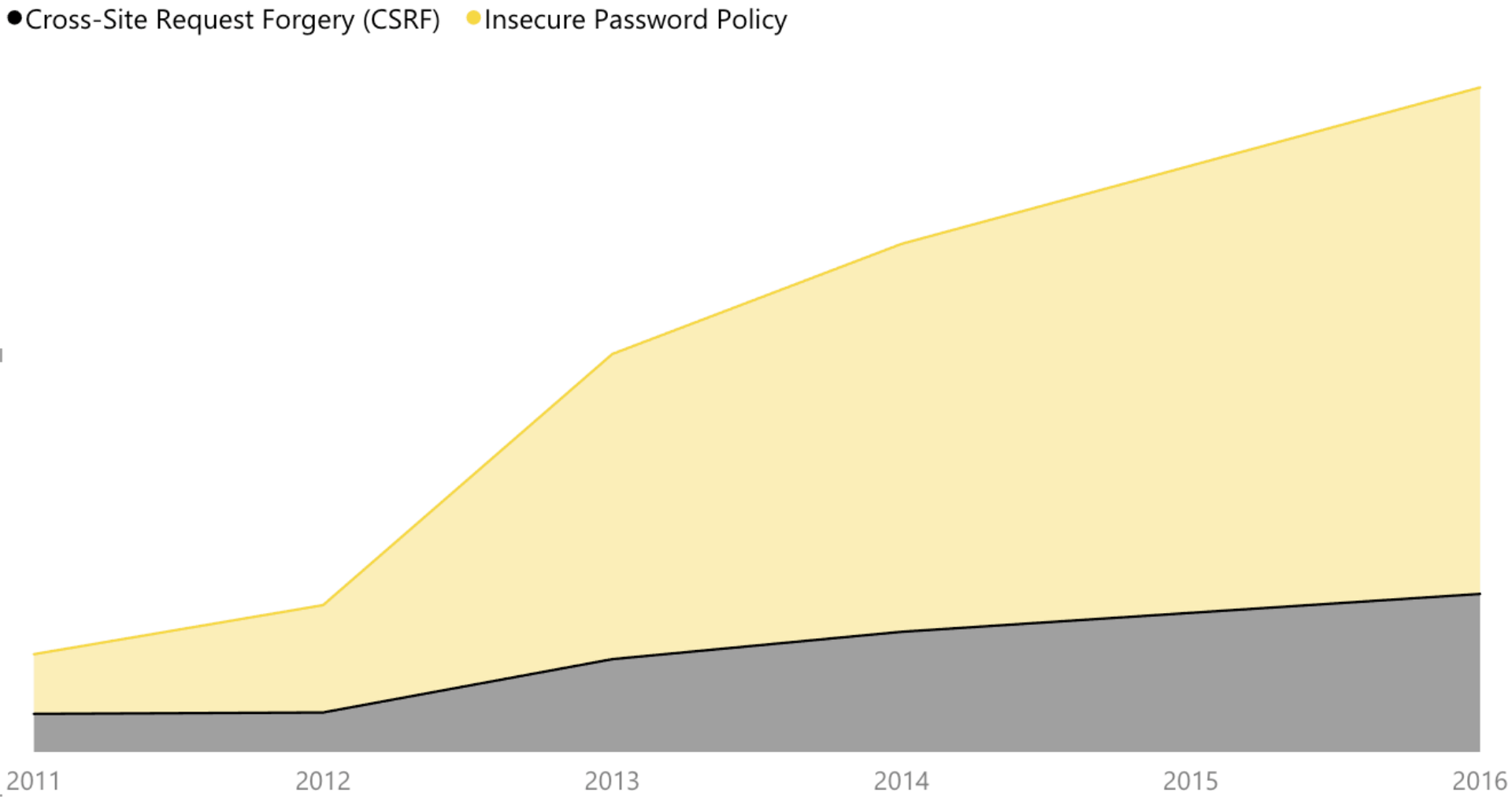
The Forgotten Killers

Description

- Very **frequently found and overlooked**
- **Trivial** to exploit
- Insecure Password Policy
  - Having a **weak password policy** in place
  - Often associated with **insecure password storage**
- Cross-site Request Forgery (CSRF)
  - Processing **requests** that where **not sourced from the application**

Trustwave® SpiderLabs®

The Forgotten Killers



Insecure Password Policy

- Why is an insecure password policy and insecure password storage bad?
- Let's review some examples…

Adobe – 153 Million Accounts (2013)

- Username, password and password hint compromised
- Improper password storage
- Most passwords cracked within days and made publicly available

Bell Canada – 22 Thousand Accounts (2014)

- Affected Bell small business customers
- Email, username, password credit card data compromised
- Improper password storage

Comcast – 590 Thousand Accounts (2015)

- Data was being sold on the underground market
- Email and passwords compromised
- Improper password storage

Last.fm – 43 Million Accounts (2012)

- Full extent was not known publicly until 2016
- Email, username and passwords were compromised
- Passwords hashed with MD5 and not salted
- Most passwords were easily cracked
- Most used password: “123456”

LinkedIn – 164 Million Accounts (2012)

- Hacked in 2012 but found on a dark market in 2016
- Email and passwords were compromised
- Hashed with SHA1 and no salt
- Most passwords were easily cracked

Cross-Site Request Forgery

- Why is cross-site request forgery bad?
- Let's review some examples…

PayPal CSRF

- Affected account management
- Allowed the attacker to add/remove/confirm email, change security questions, add full privilege users to business accounts, etc.
- Publicly disclosed in 2014

Go Daddy CSRF

- Affected domain registrations
- Allowed an attacker to hijack a victim's domain
- Publicly disclosed in 2015

Hilton CSRF

- Affected the Hilton Honors loyalty program
- Allowed an attacker to hijack a victim's account
- Publicly disclosed in 2015

Too Many Network Device Affected by CSRF

- Examples:
  - Netgear Routers
  - Cisco Residential Gateway
  - Siemens Ruggedcom NMS
  - Huawei 3G Router
  - Ubiquiti Networking Products
- Impact is often full control and administration of the device
- Often used for botnets

Xzeres CSRF

- Affects 442SR Wind Turbine
- Allows an attacker to cut off power to ALL attached systems
- Publicly disclosed in 2015

The Forgotten Killers

Why they matter

- Your **firewall, IDS, IPS and/or WAF will not help you**
- These vulnerabilities will **put you in the news**
- **Reputational damage**
- Insecure Password Policy and Insecure Password Storage
  - If you allow it, **most users will have a weak password**
  - Users **reusing passwords compounds the problem**
- Cross-Site Request Forgery (CSRF)
  - Victim issues requests against vulnerable site without knowing it
  - **Often combined with other attacks** (i.e.: spear phishing attacks)
  - Easy to find and **trivial** to exploit

Trustwave® SpiderLabs®

The Forgotten Killers

Mitigation

- Insecure Password Policy
  - Define a **solid password policy**
  - Ensure all application **enforces that policy**
  - Ensure passwords are **properly salted and hashed** (bcrypt)
- Cross-site Request Forgery
  - Use **anti-forgery token**
  - Leverage your **development framework**

Trustwave® SpiderLabs®

The Forgotten Killers

Solid Password Policy

- Be at least **10 characters** long
- **Not** be based on dictionary words or the username
- Contain **at least one** character **from each** category:
  - Uppercase letters
  - Lowercase letters
  - Numbers
  - Non-alphanumeric characters

Trustwave® SpiderLabs®

The Misunderstood

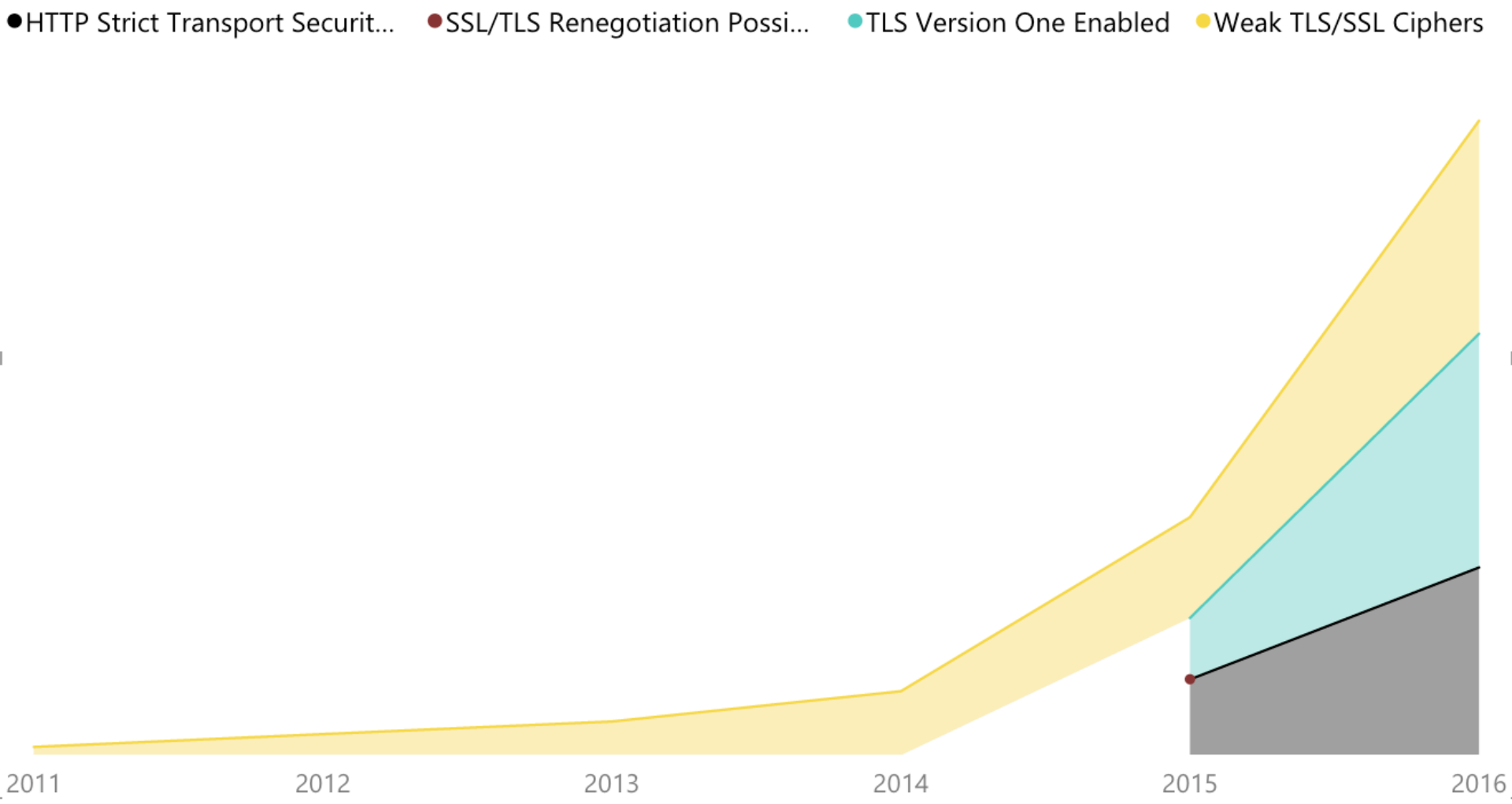
The Misunderstood

Description

- Very **frequently found and misunderstood**
- Are simple **configuration issues**
- Using Weak Ciphers
  - Using **weak ciphers** with SSL/TLS (RC4, DES)
- Using TLSv1
  - Using a **deprecated protocol** for secure communications
- Allowing TLS/SSL Renegotiation
  - **Allowing renegotiation** of configuration for an established connection
- HTTP Strict Transport Security (HSTS)
  - Tells a browser to **remember to connect over TLS** in the future

Trustwave® SpiderLabs®

The Misunderstood



The Misunderstood

Weak Ciphers – RC4

- **Designed** by Ron Rivest of RSA Security **in 1987**
- First RC4 **weakness** observed in **1995**
- Was used by **WEP on wireless networks**
- In 2001, a new attack was discovered causing a **scramble to find a new replacement** for WEP (WPA)
- **Many other weaknesses** in RC4 have been documented since
- As of 2015, its **use is prohibited in TLS per RFC 7465**

Trustwave® SpiderLabs®

The Misunderstood

Weak Ciphers – DES

- **Designed** by IBM in the early **1970's**
- First **weakness** observed in **1991**
- First **publicly broken** DES encrypted message in **1997** (DESCHALL)
- Feat **repeated in 1998 and 1999** with best time of **22 hours and 15 minutes**
- 56-bit **key size is too small**
- DES was then **replaced with 3DES**

Trustwave® SpiderLabs®

The Misunderstood

TLSv1

- In **2014**, a vulnerability in SSL 3.0 was identified by Google and named **POODLE** (Padding Oracle On Downgraded Legacy Encryption)
- POODLE allows an attacker to **steal information from a secure connection** such as ”secure” HTTP cookies, HTTP authorization headers, etc.
- A **few months later**, certain implementations of **TLSv1 were also found to be vulnerable** to the POODLE attack
- TLSv1 is also **vulnerable to the CRIME and BEAST** attacks
- The PCI Security Standards Council requires that **TLSv1 be retired by June of 2018**

Trustwave® SpiderLabs®

The Misunderstood

SSL/TLS Renegotiation

- Worst case, allows attacker to **inject data in a victim's session** to execute requests
- Best case, allows an attacker to **create a denial of service condition** (DoS attack)
  - SSL/TLS supports both **client and server initiated** as well as **secure and insecure renegotiation**
  - A **client can initiate a secure renegotiation** which uses around **15 times more resources** on the server end
  - A **tool as been published** to make this attack simple

Trustwave® SpiderLabs®

The Misunderstood

HTTP Strict Transport Security

- Introduced in **2012** as part of RFC 6797
- Tells a **browser to always request the domain over HTTPS** for a certain amount of time (i.e.: 6 months)
- Goal is to **prevent initial client interaction over HTTP** and redirection to HTTPS
- **Minimizes the chances for a MITM attack** to occur during the redirection phase
- Not only to protect content in transit but **also to prevent interception, injection and tampering** of traffic

Trustwave® SpiderLabs®

The Misunderstood

Why they matter

- High probability they are **found in many areas of your environment**
- Will **protect you against more skilled and/or determined attackers**
- They are **easier to fix**
- Will keep you **in compliance or ahead of the curve** for upcoming compliance requirements

Trustwave® SpiderLabs®

The Misunderstood

Mitigation

- Requires **configuration changes** at the load balancer, WAF and/or web server level
- Usually involves **no or very limited code changes**
- Mitigation procedure is **dependent on devices in use** in the environment
- **HSTS** is configured by adding a **"Strict-Transport-Security" header** to the application's responses.

Trustwave® SpiderLabs®

The Personal Problems

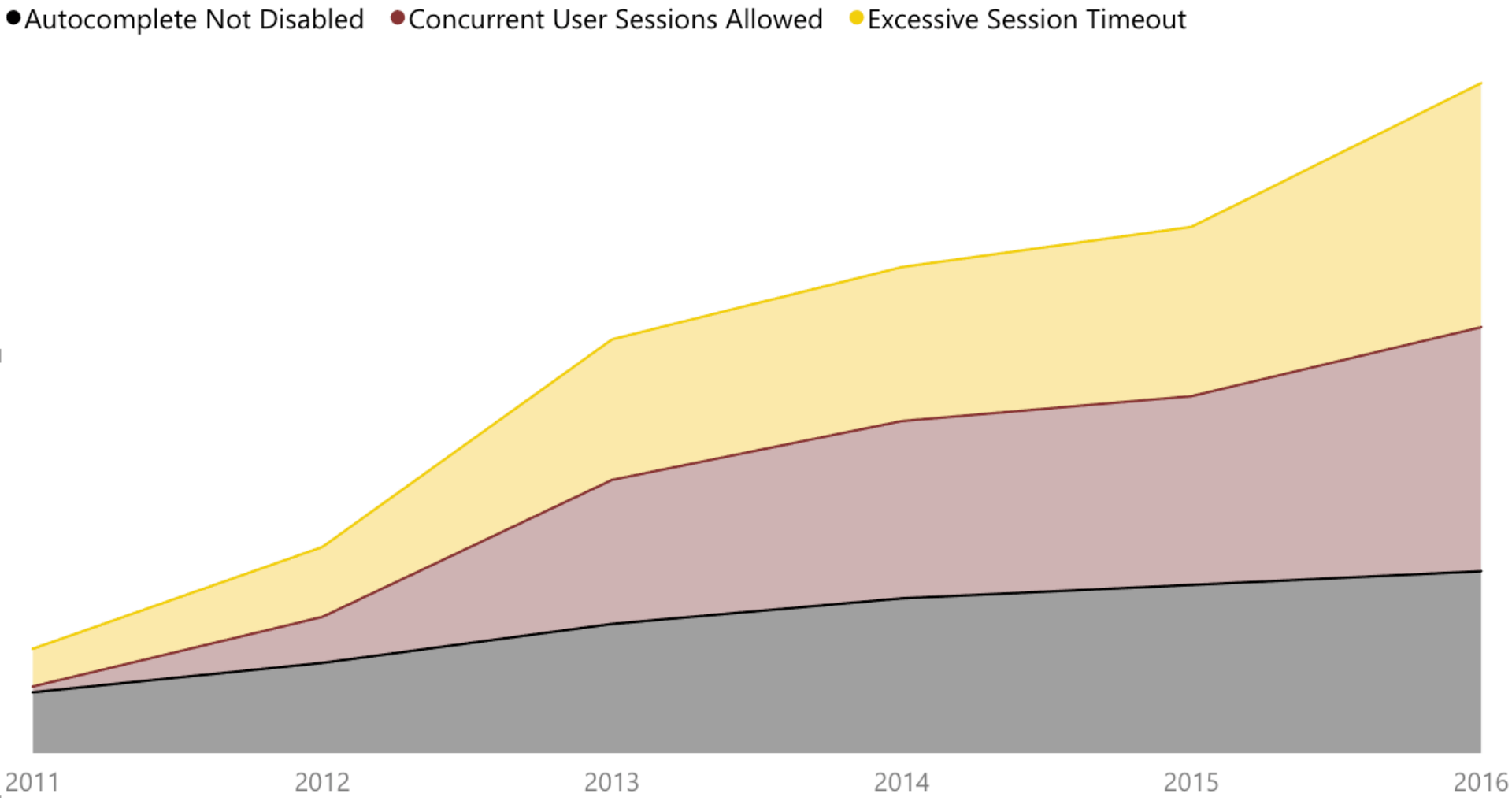
The Personal Problems

Description

- Very **frequently found** and most often **affect internal users**
- Autocomplete not disabled
  - Allowing **browsers to save passwords** and other sensitive information
- Concurrent user sessions allowed
  - Allowing the **same user** to be authenticated in **two different browser sessions**
- Excessive session timeout
  - Allowing a **user session** to be **active for a long duration**

Trustwave® SpiderLabs®

The Personal Problems



The Personal Problems

Why they matter

- Autocomplete not disabled
  - Given physical access, an **attacker** will be able to **access the protected web application**
  - When compromised over the network, an **attacker** may be able to **retrieve the saved passwords** from the browser
- Concurrent user sessions allowed
  - Encourages **account sharing**
    - May prevent **ability to attribute action** to a user
- Excessive session timeout
  - Given physical access, an **attacker** may be able to **access the protected web application**
  - When compromised over the network, an **attacker** may be able **to leverage an active session**

Trustwave® SpiderLabs®

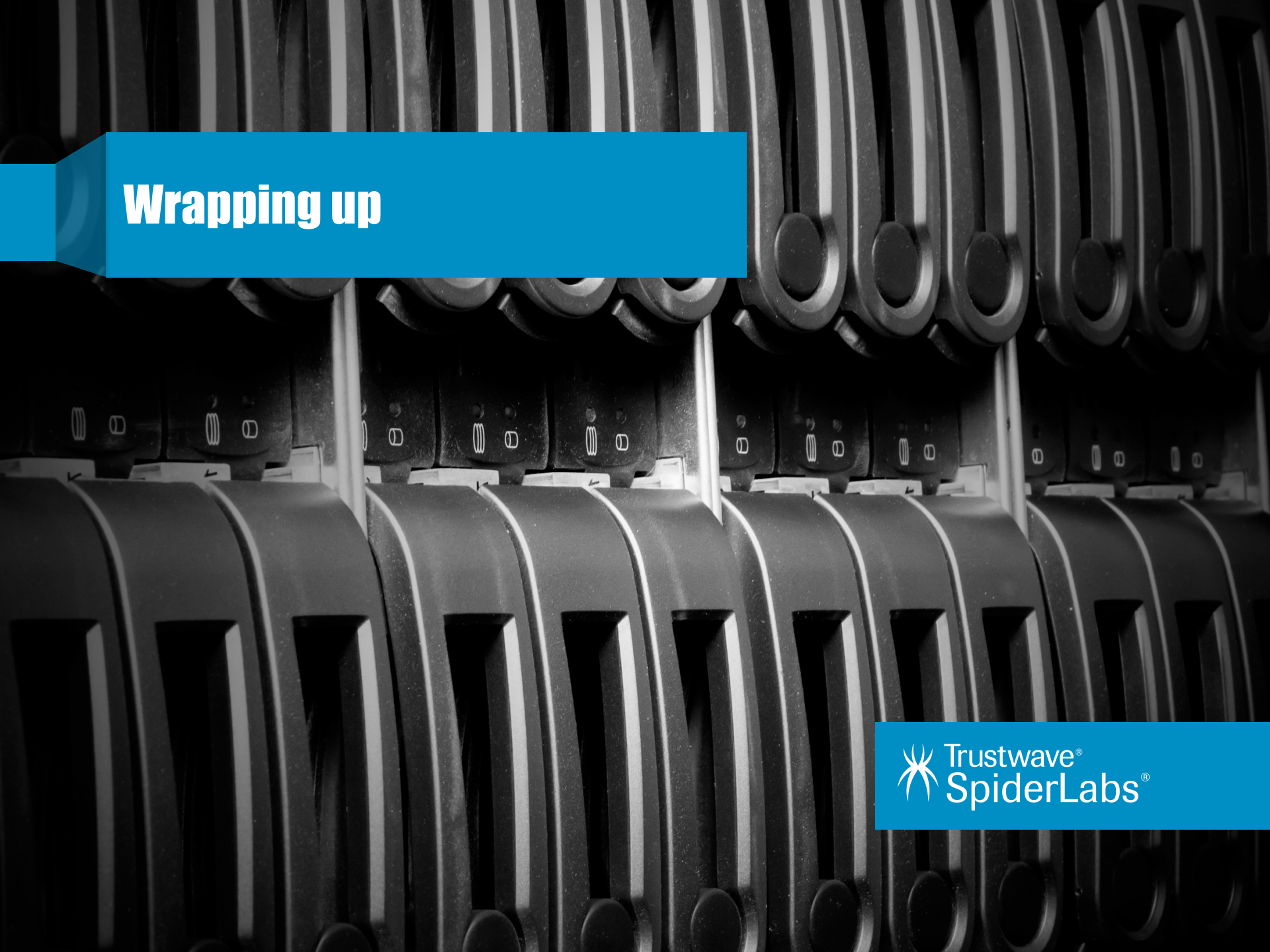
The Personal Problems

Mitigation

- Autocomplete not disabled
  - Usually a **simple code change**
- Concurrent user sessions allowed
  - Usually fixed **at the application level**
- Excessive session timeout
  - Usually a **configuration change** at the application or web server level

Trustwave® SpiderLabs®

Wrapping up



Trustwave® SpiderLabs®

Recap

- **SQLi and XSS** is **still strong** but slightly **on the decline**
- The **upcoming** web application vulnerability **stars** appear to be based **around authentication and authorization vulnerabilities**
- Organizations are still **struggling to properly configure SSL/TLS**
- There are a **lot of easy to fix and commonly found issues** that organizations should **prioritize**
- **Do not only fix severe vulnerabilities**, smaller ones will get you too
- There are **more low tech bad guys than elite hackers** out there trying to get to your data so **fixing lower severity issues is critical**

Trustwave® SpiderLabs®

Final Words

- Hopefully I helped you **identify areas of your web applications** that could use more attention
- You **have to pay for security** regardless
  - **Upfront is cheaper** than later
  - **Security debt accumulates** just like other technical debt
  - **Invest** in the security of your web applications
- Information security **affects** not only your organization but **all of us** as well
- Do your **due diligence** by following **best practices** so you don't have to be **another news story** and can stay **ahead of the curve** when **new regulations** for your industry are introduced

Trustwave® SpiderLabs®

Questions?