Understanding Web Application Threats and Mitigations

Nick Russo
NETWORK ENGINEER

@nickrusso42518 www.njrusmc.net

Agenda

- Top OWASP threats
- Exploring the injection attack
- Secret and general data management
- External devices for extra security

What Is OWASP?

Open Web Application Security Project

Provides free resources

Popular "Top 10" vulnerability list

OWASP Top Ten Reference

OWASP Top 10 - 2013		OWASP Top 10 - 2017
A1 – Injection	→	A1:2017-Injection
A2 – Broken Authentication and Session Management	→	A2:2017-Broken Authentication
A3 – Cross-Site Scripting (XSS)	↘	A3:2017-Sensitive Data Exposure
A4 – Insecure Direct Object References [Merged+A7]	∪	A4:2017-XML External Entities (XXE) [NEW]
A5 – Security Misconfiguration	↘	A5:2017-Broken Access Control [Merged]
A6 – Sensitive Data Exposure	↗	A6:2017-Security Misconfiguration
A7 – Missing Function Level Access Contr [Merged+A4]	∪	A7:2017-Cross-Site Scripting (XSS)
A8 – Cross-Site Request Forgery (CSRF)	☒	A8:2017-Insecure Deserialization [NEW, Community]
A9 – Using Components with Known Vulnerabilities	→	A9:2017-Using Components with Known Vulnerabilities
A10 – Unvalidated Redirects and Forwards	☒	A10:2017-Insufficient Logging&Monitoring [NEW,Comm.]

https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project

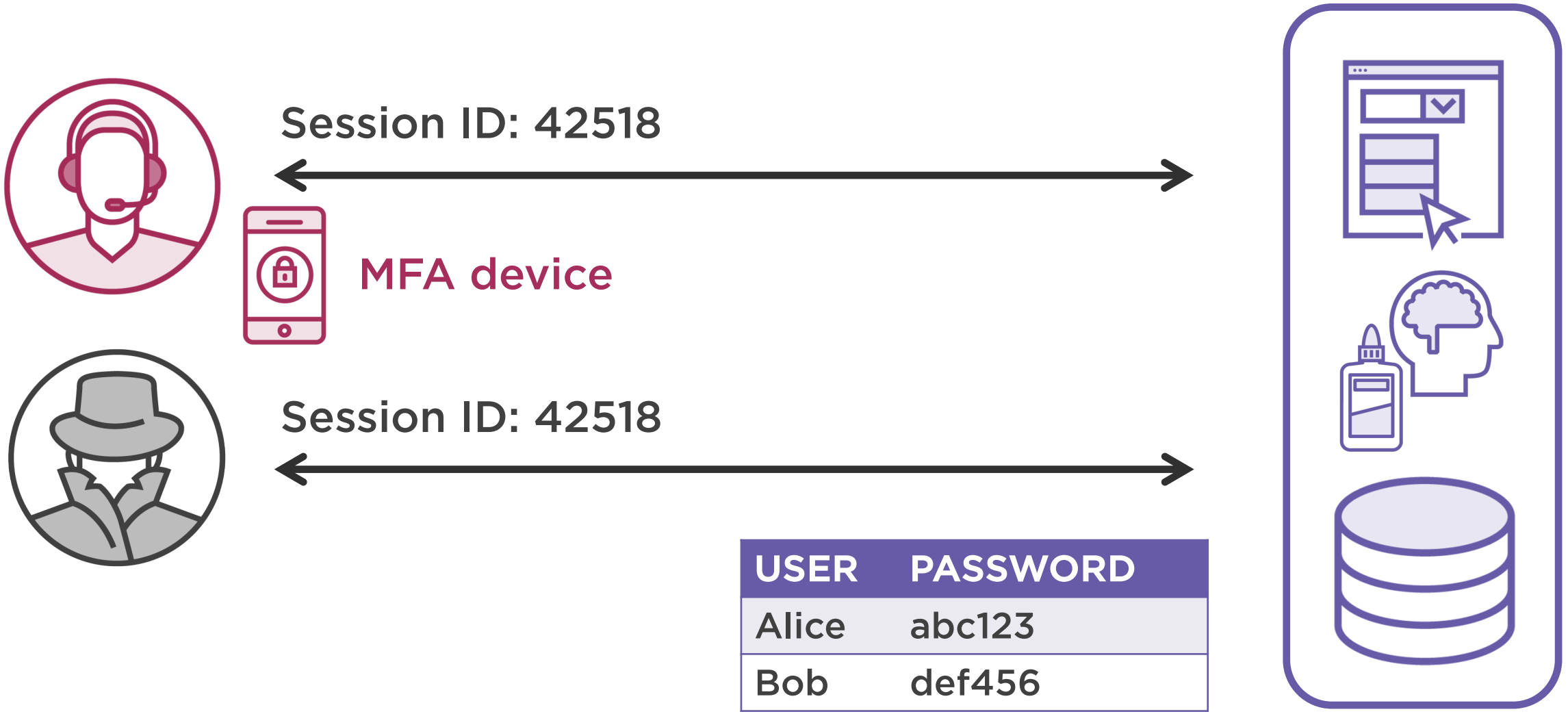
Injection Attacks

HTTP POST

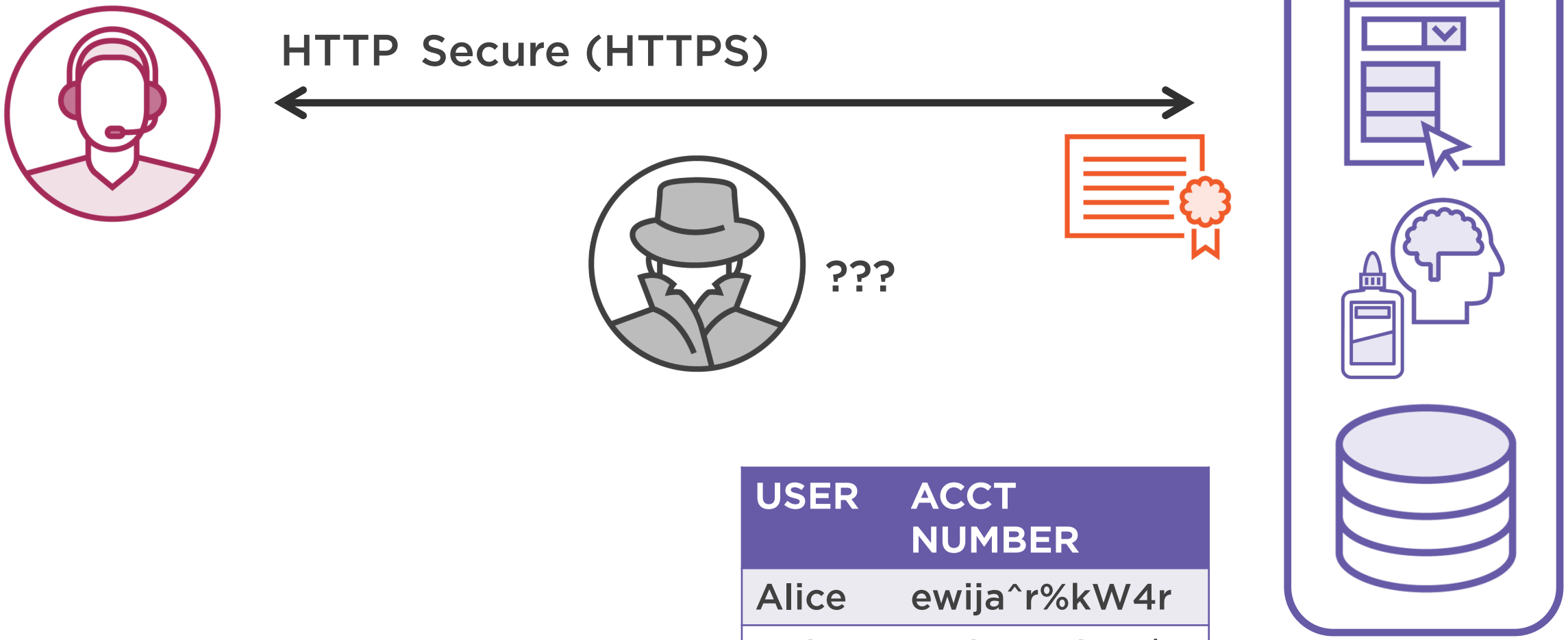
```
<subject>Don't mind me!</subject>
<body>(code to insert "hacker" user)</body>
```

USER	PASSWORD
Alice	abc123
Bob	def456
hacker	just_added

Broken Authentication



Sensitive Data Exposure



USER	ACCT NUMBER
Alice	ewija^r%kW4r
Bob	8jEbH4(d!Xv$p

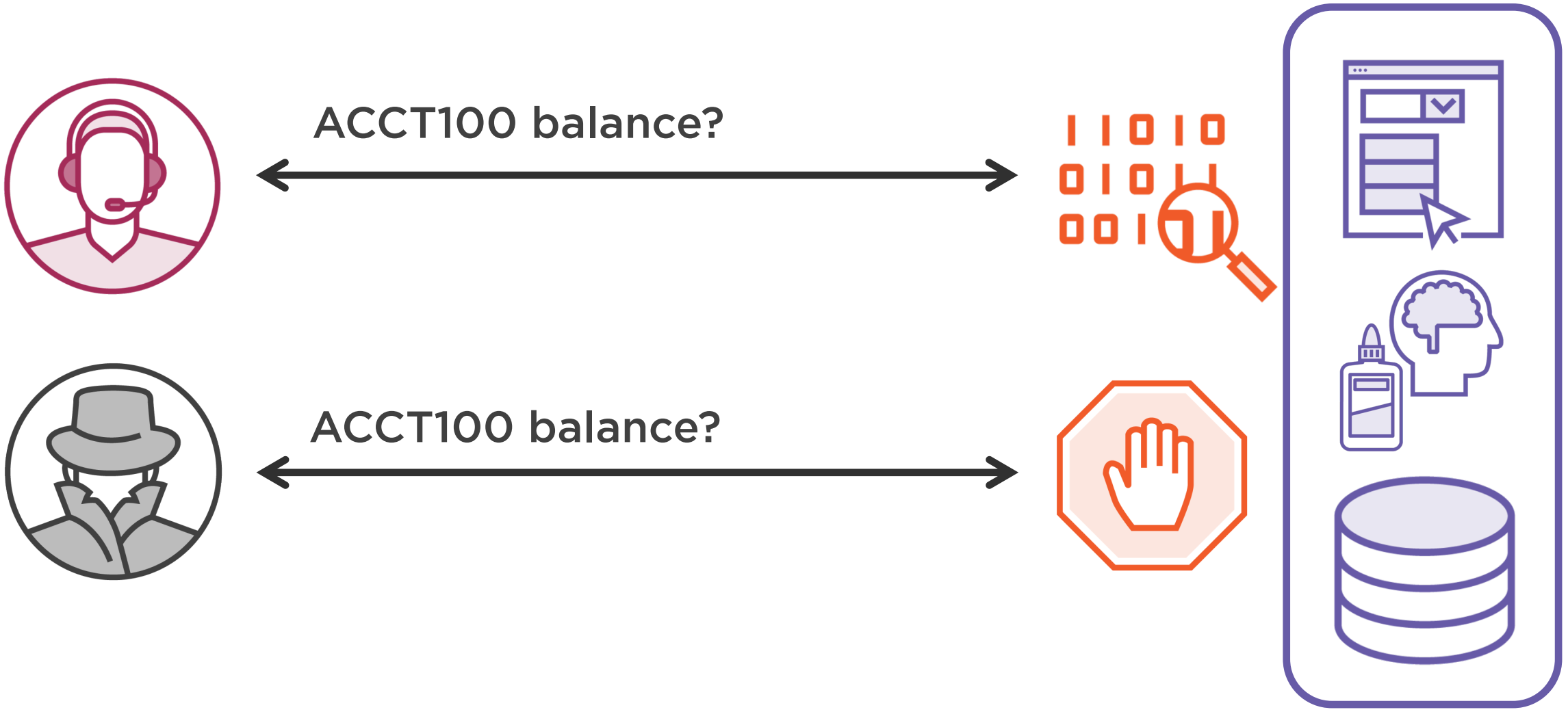
XML External Entity (XXE)

HTTP POST

```
<!DOCTYPE attack [
<!ENTITY pwd SYSTEM "file:/etc/passwd">
]>
<body>
  <data>/etc/passwd is &pwd;</data>
</body>
```

User Data

Broken Access Control



Security Misconfigurations

Vulnerability scan

SERVICE	USER	PASSWORD
Web server	admin	tomcat
SQL server	sa	Password123

Cross-site Scripting (XSS)

HTTP PUT

```
<script type="text/javascript">
  alert(document.cookie);
</script>
```

HTTP GET

"Why is my session data popping up in front of me?"

Cross-site Request Forgery (CSRF)

Insecure Deserialization

"$0 balance, I'm debt free!"

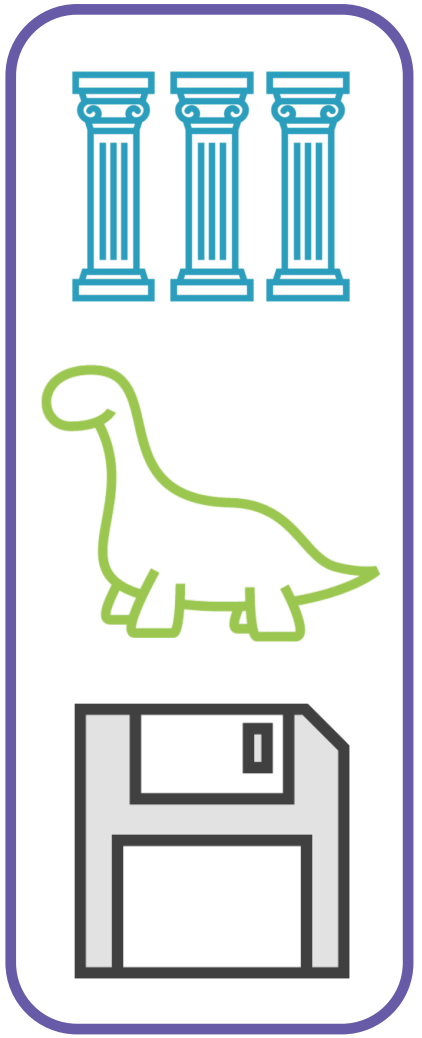
```
acct = deserialize(bytes)
acct.set_balance(100000)
bytes = acct.serialize()
```

Bob's balance: $0.00

Bob's balance: $100,000.00

Using Components with Known Vulnerabilities

Vulnerability scan



"We updated the system in 1998, right now the business is focused on growing sales, not IT."

Insufficient Logging and Monitoring

Highly-skilled attacker breaches

Continuous monitoring (SIEM)

Demo

Exploring the injection attack

```
import getpass
secret = getpass.getpass()
print(
    type(secret),
    secret
)
```

- Comes standard with Python
- Prompt for password using OS mechanism; text not revealed
- A bad idea just for illustration

```
$ python getpasstest.py
Password:
<class 'str'> d3v0p$
```

- Run the program
- Reflect type and value

```
import os
secret = os.environ["SECRET"]
print(
    type(secret),
    secret
)
```

- Comes standard with Python
- Read the env variable "SECRET"
- A bad idea just for illustration

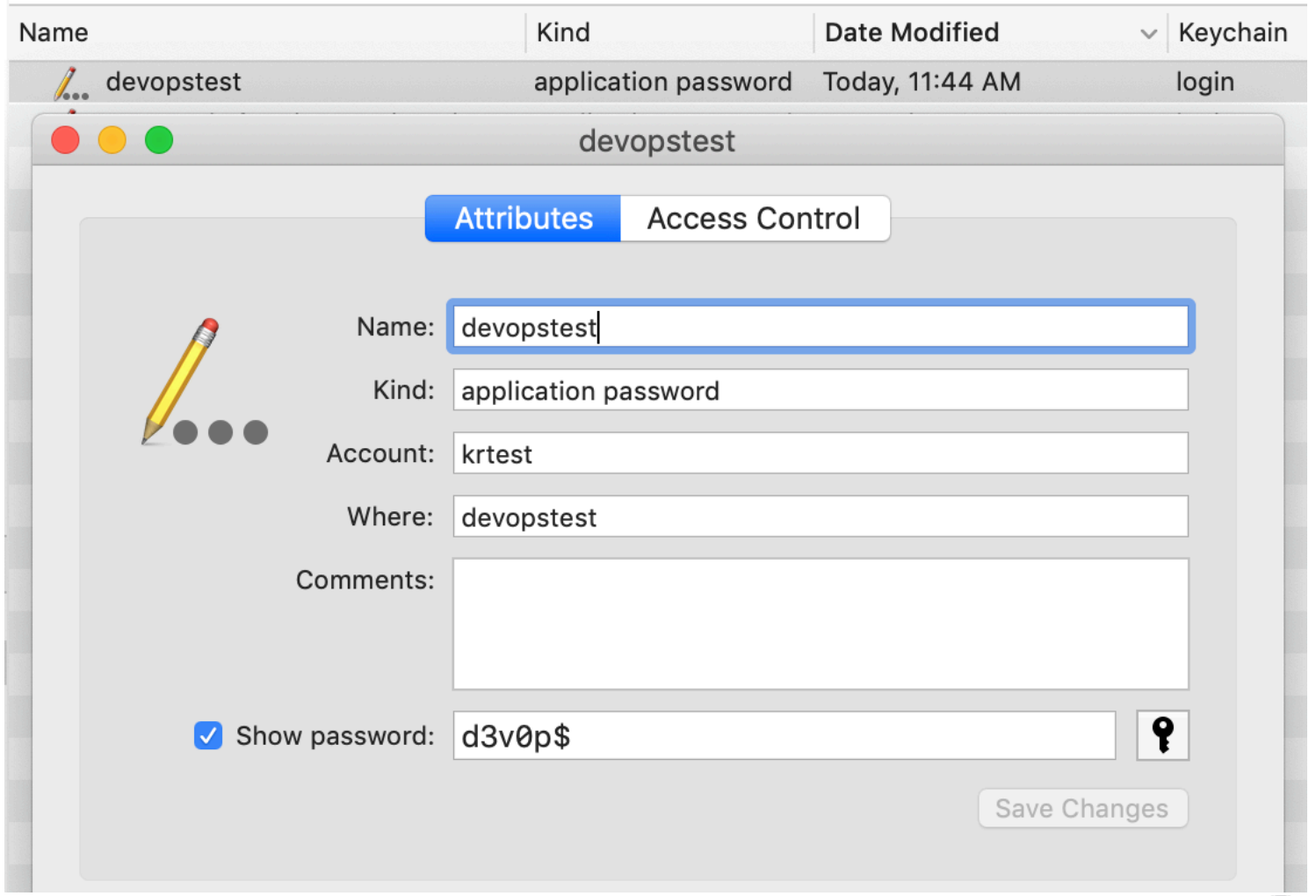
```
$ export SECRET="d3v0p$"
$ python3 envtest.py
<class 'str'> d3v0p$
```

- Define "SECRET" env variable
- Run the program
- Reflect type and value

Python "keyring" Library

```
$ pip install keyring

import keyring
keyring.set_password(
    "devopstest",
    "krtest",
    "d3v0p$"
)
```



```
$ cat krtest.py
import keyring
secret = keyring.get_password(
    "devopstest",
    "krtest"
)
print(type(secret), secret)
```

- Read the keyring password "devopstest" into "secret"
- A bad idea just for illustration

```
$ python3 krtest.py
<class 'str'> d3v0p$
```

- Run the program
- Reflect type and value

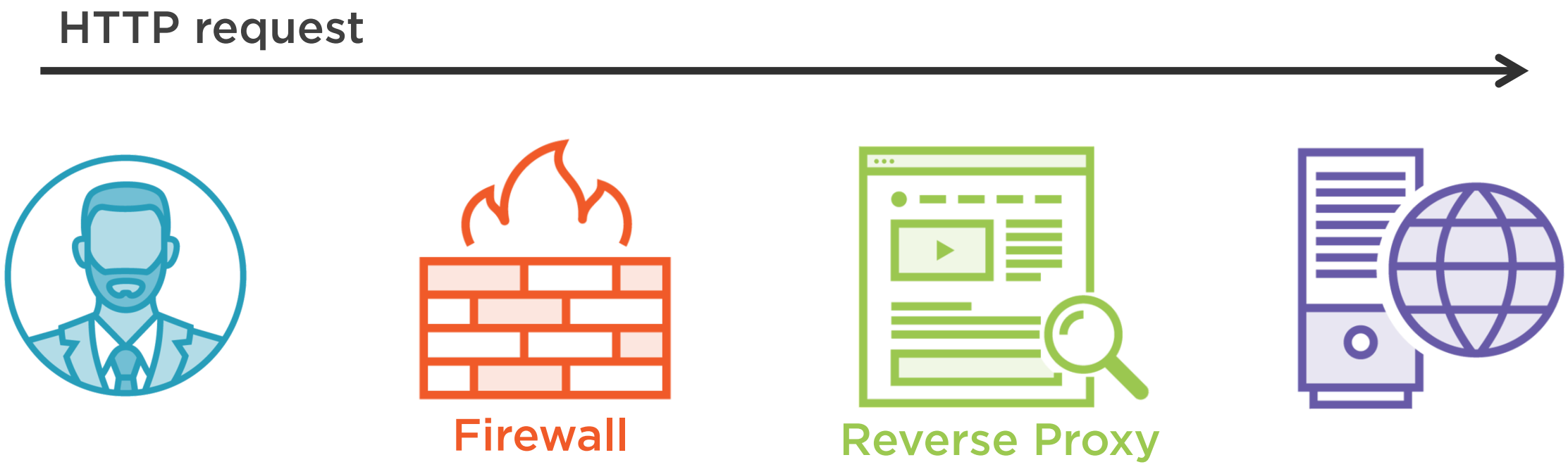
Other Approaches

Restricted plain text files

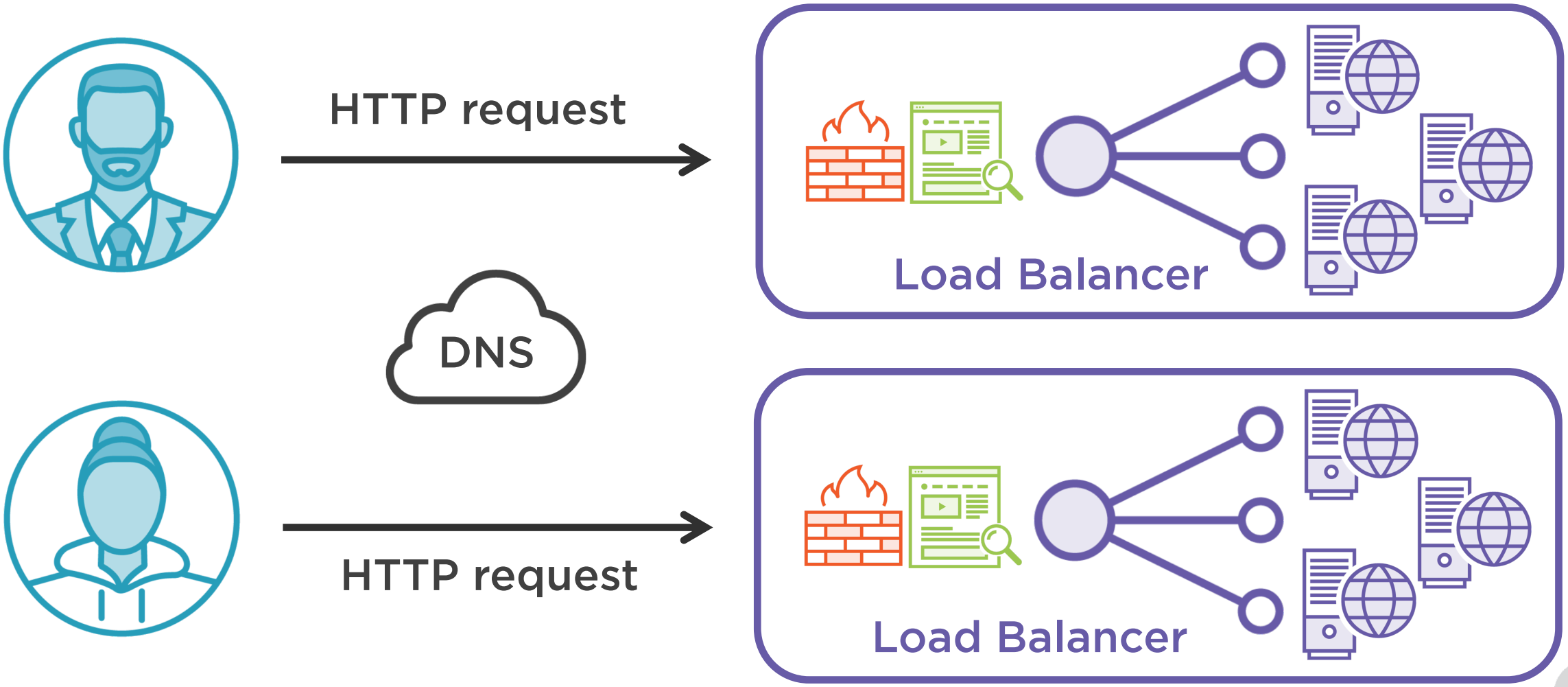
Encrypted vault files

Many commercial products

Firewall and Reverse Proxy Tag Team



Load Balancers and DNS; Achieving Scale



Summary

So many products

So many design considerations

Continue your education

Thank you!