Dark Clouds ahead: Attacking a Cloud Foundry Implementation

Pablo Artuso, Nahuel D. Sánchez

March 2019 - TROOPERS 19, Heidelberg - Germany

TROOPERS

onapsis

Disclaimer

This presentation contains references to the products of SAP SE. SAP, R/3, xApps, xApp, SAP NetWeaver, Duet, PartnerEdge, ByDesign, SAP Business ByDesign, and other SAP products and services mentioned herein are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world.

Business Objects and the Business Objects logo, BusinessObjects, Crystal Reports, Crystal Decisions, Web Intelligence, Xcelsius and other Business Objects products and services mentioned herein are trademarks or registered trademarks of Business Objects in the United States and/or other countries.

SAP SE is neither the author nor the publisher of this publication and is not responsible for its content, and SAP Group shall not be liable for errors or omissions with respect to the materials.

Who are we?

Pablo Artuso

Nahuel D. Sánchez

- Focus on vulnerability research and ERP security
- Trainers & speakers at Security conferences
- Frequent vulnerability reporters for SAP products
- *https://www.onapsis.com*

Agenda

- Cloud Foundry
- SAP HANA and XSA
- Our Research
- Conclusions

Cloud foundry

Introduction

Cloudfoundry?

"...It is an open source platform that you can deploy to run your apps on your own computing infrastructure, or deploy on an IaaS like AWS, vSphere, or OpenStack…"

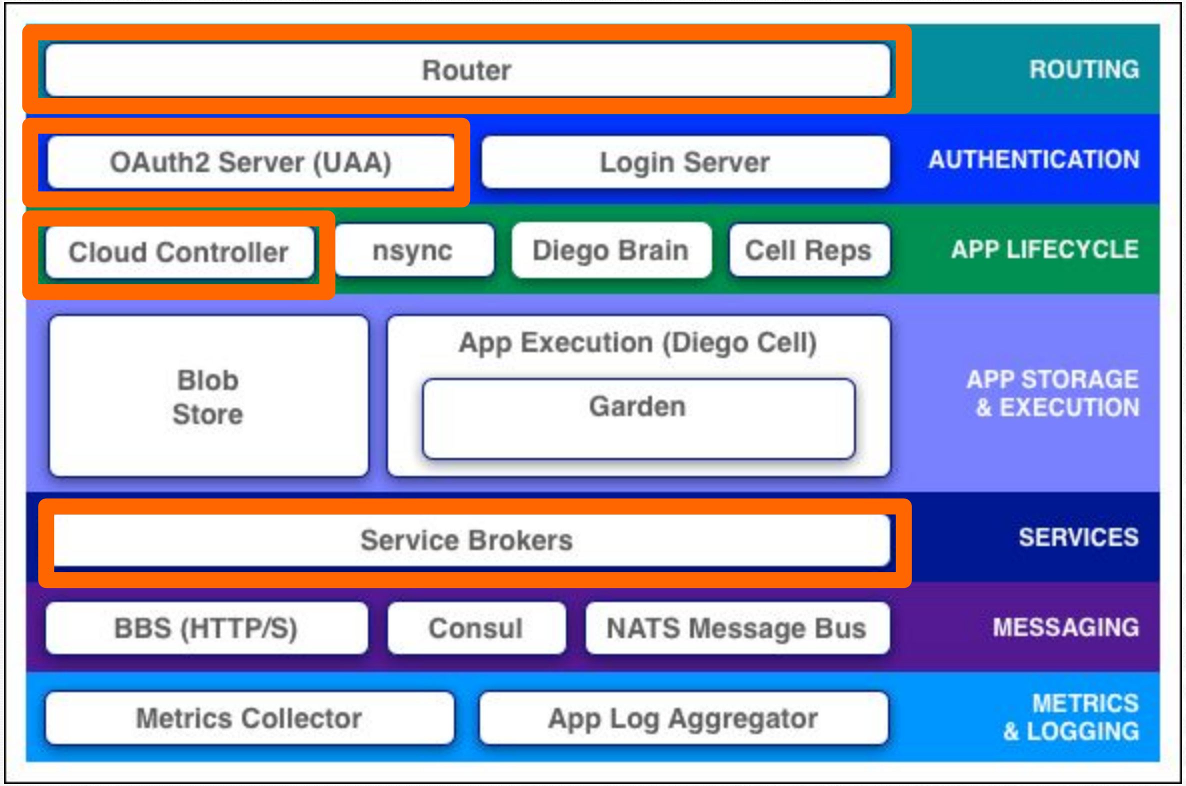
- Abstraction
- Scalability
- Simplicity of use

https://docs.cloudfoundry.org/concepts/overview.html

Not all clouds are equal

Cirrocumulus
Cirrus
Altocumulus
Altostratus
Stratocumulus
Stratus
Cumulus
Cumulonimbus

Cloud foundry components



Routes traffic to the appropriate component

Provides Access to 3rd party services to deployed applications

- Database Access
- File system Storage
- Others…

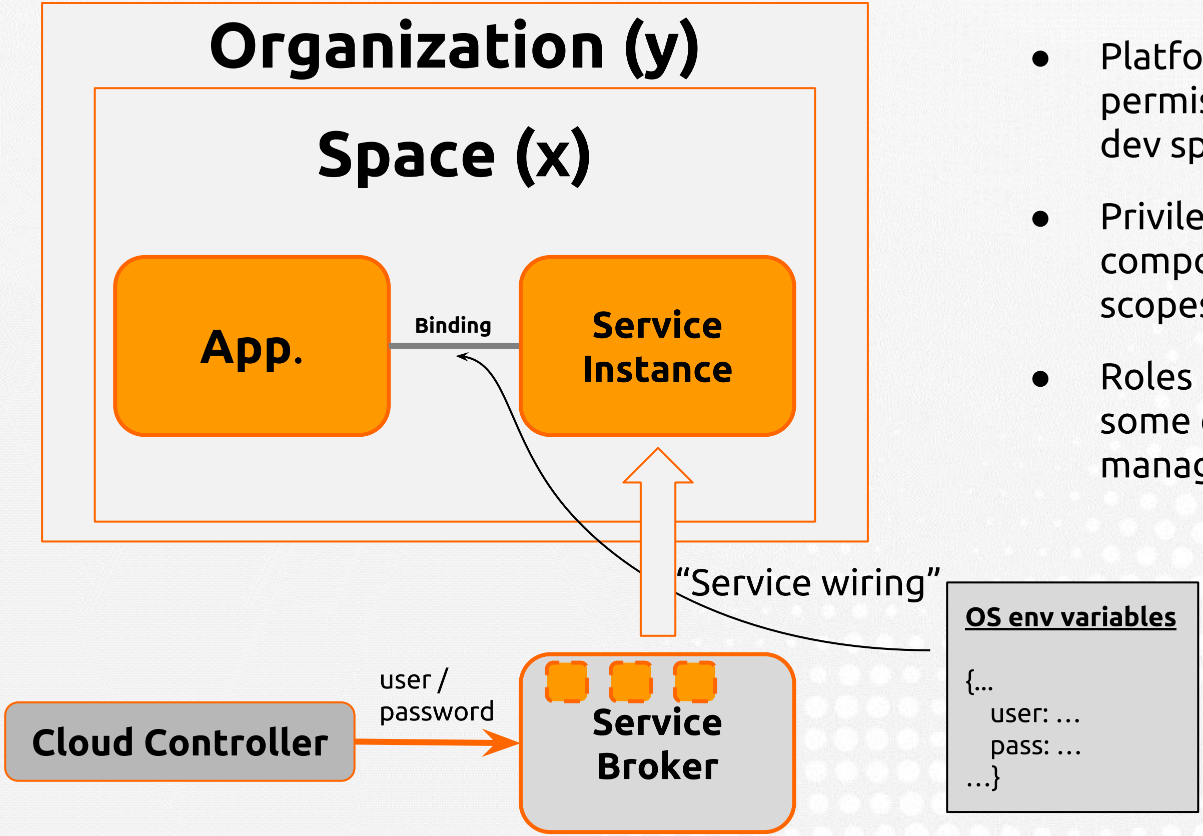
Provides user authentication services

Orchestrates application deployment and maintains track of organizations, user roles and other…

Cloud foundry implementations

Pivotal, Atos, Cloud.gov, IBM, **SAP**, others…

Cloudfoundry 101 (Concepts)



OS env variables

```
{...
   user: ...
   pass: ...
...}
```

- Platform users can have different permissions in different spaces (think in dev space/prod space)
- Privileges are grouped into "scopes", each component have their our scopes (UAA, scopes, Controller Scopes, and so on)
- Roles are a group of scopes. There are some default roles such as: Admin, Org manager, Auditor and others...

https://docs.cloudfoundry.org/concepts/roles.html
https://help.sap.com/viewer/4505d0bdaf4948449b7f7379d24d0f0d/2.0.00/en-US/df19a03dc07e4ba19db4e0006c1da429.html

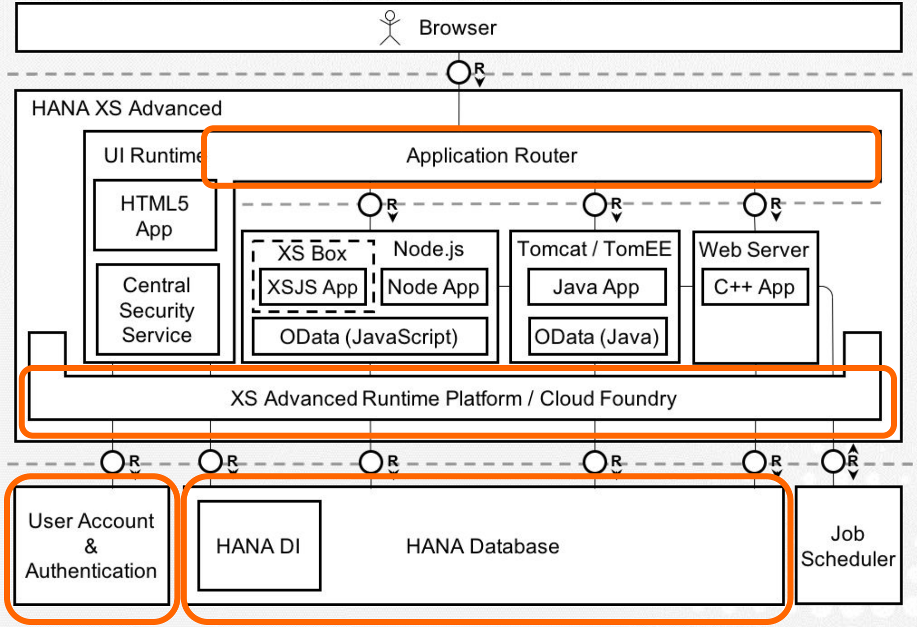
SAP HANA and XSA

SAP HANA, XS Advanced overview

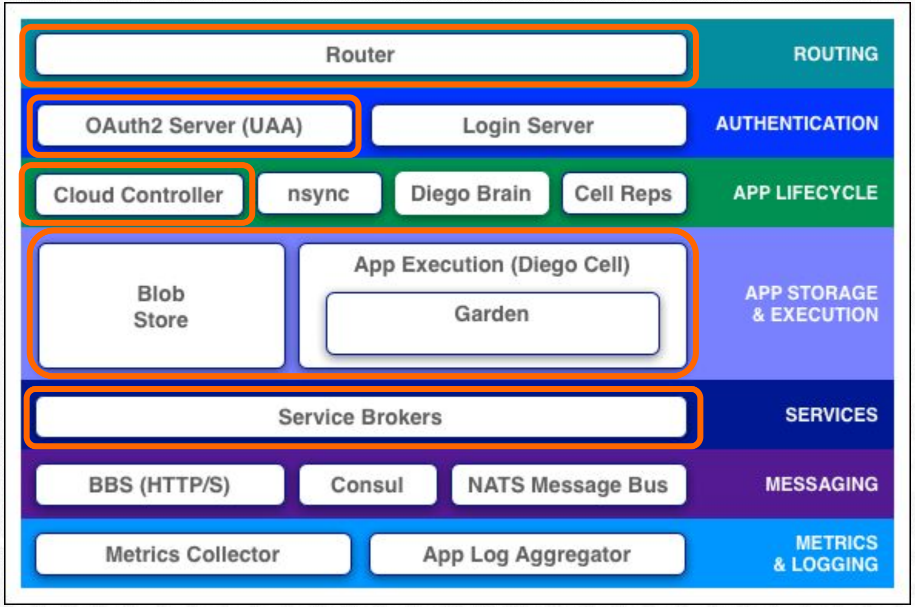
- SAP HANA?
  - In memory database
- XS Engine (Classic)
  - Embedded web server (now deprecated)
- Application platform
  - XS Advanced
  - In the cloud, provided by Cloud Foundry
  - On premise, provided by SAP (Cloud Foundry compatible)

XS Advanced <-> Cloud foundry

XSA



Cloud foundry



XS Advanced network connections

Client	Service	Port (dest)	Usage
Applications UI	xsuaaserver	3xx32	Client App -> XS Web disp **(Auth)**
CMD cli / Cli libs / others	xscontroller	3xx30	Connections to the xscontroller **(Data access)**
Applications UI	App Instances	51000-51500	End users access Applications
App ports 51000-51500	App Instances	50000-50999	Connection from web disp to the target
xsexecagent	xscontroller	3xx29	Connections from xs exec agent and the controller

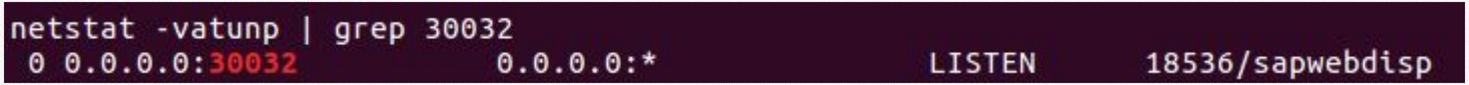
Legend:

Public ports

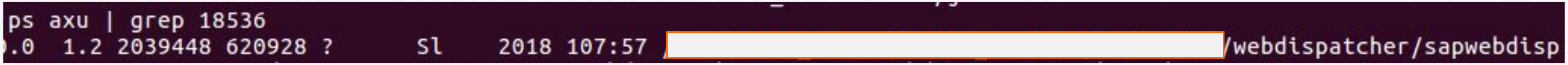
Internal ports

https://testhelpportal.com/viewer/6b94445c94ae495c83a19646e7c3fd56/2.0.03/en-US/6b039ca5e9e44402bd86cafa53ea850b.html

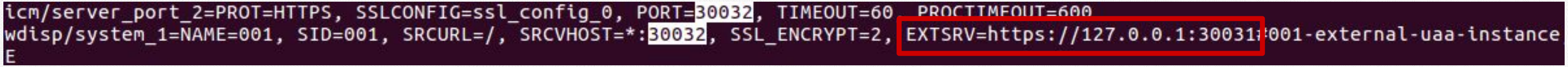
XS Advanced network connections (example)



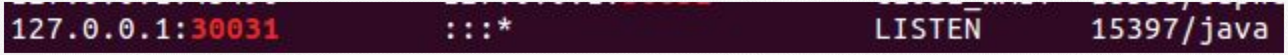
```
netstat -vatunp | grep 30032
 0 0.0.0.0:30032           0.0.0.0:*                LISTEN      18536/sapwebdisp
```



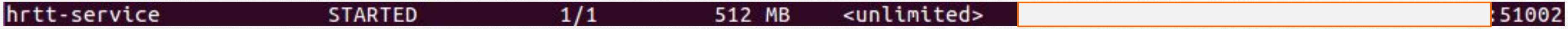
```
ps axu | grep 18536
.0   1.2 2039448 620928 ?        Sl     2018 107:57 [redacted]/webdispatcher/sapwebdisp
```



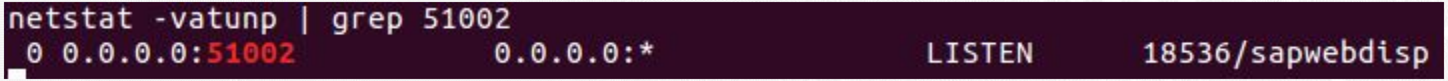
```
icm/server_port_2=PROT=HTTPS, SSLCONFIG=ssl_config_0, PORT=30032, TIMEOUT=60, PROCTIMEOUT=600
wdisp/system_1=NAME=001, SID=001, SRCURL=/, SRCVHOST=*:30032, SSL_ENCRYPT=2, EXTSRV=https://127.0.0.1:30031#001-external-uaa-instance
E
```



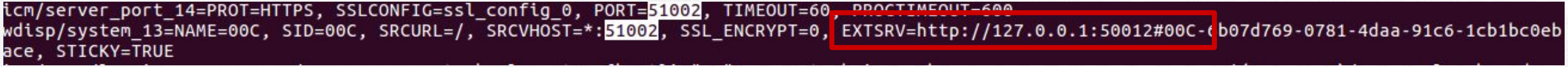
```
127.0.0.1:30031          :::*                    LISTEN      15397/java
```



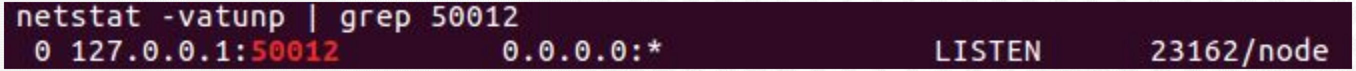
```
hrtt-service                STARTED          1/1         512 MB     <unlimited>  [redacted]:51002
```



```
netstat -vatunp | grep 51002
 0 0.0.0.0:51002           0.0.0.0:*                LISTEN      18536/sapwebdisp
```



```
icm/server_port_14=PROT=HTTPS, SSLCONFIG=ssl_config_0, PORT=51002, TIMEOUT=60, PROCTIMEOUT=600
wdisp/system_13=NAME=00C, SID=00C, SRCURL=/, SRCVHOST=*:51002, SSL_ENCRYPT=0, EXTSRV=http://127.0.0.1:50012#00C-6b07d769-0781-4daa-91c6-1cb1bc0eb
ace, STICKY=TRUE
```

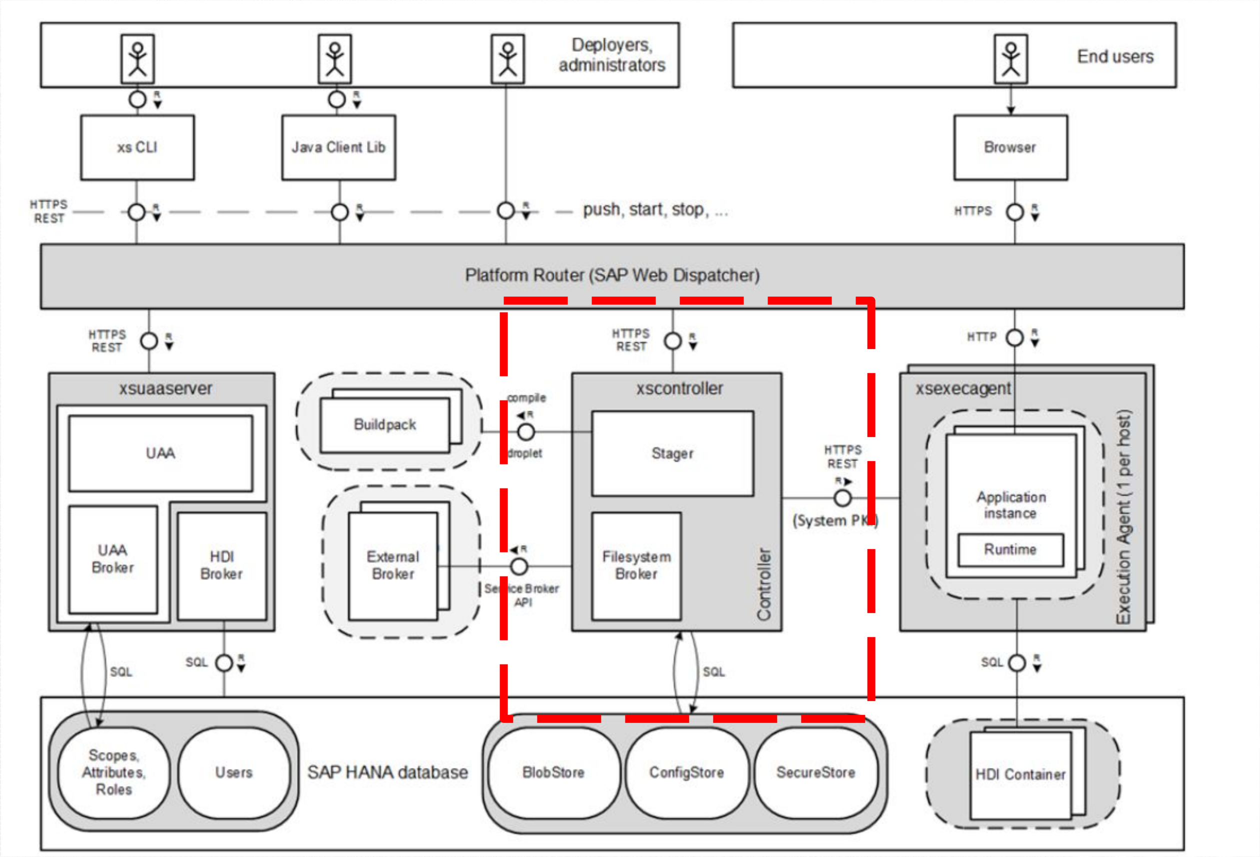


```
netstat -vatunp | grep 50012
 0 127.0.0.1:50012         0.0.0.0:*                LISTEN      23162/node
```

Our research

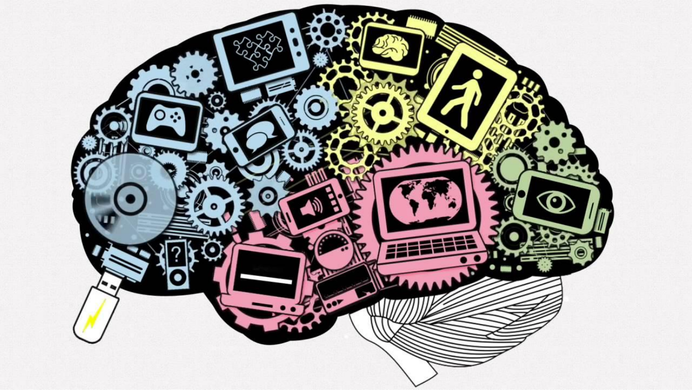
onapsis

Focus


Deployers, administrators
End users
xs CLI
Java Client Lib
Browser
HTTPS REST
push, start, stop, ...
HTTPS
Platform Router (SAP Web Dispatcher)
HTTPS REST
HTTPS REST
HTTP
xsuaaserver
UAA
UAA Broker
HDI Broker
Buildpack
compile
droplet
External Broker
Service Broker API
xscontroller
Stager
Filesystem Broker
Controller
HTTPS REST
(System PKI)
xsexecagent
Application instance
Runtime
Execution Agent (1 per host)
SQL
SQL
SQL
SQL
Scopes, Attributes, Roles
Users
SAP HANA database
BlobStore
ConfigStore
SecureStore
HDI Container

Who is the Controller?

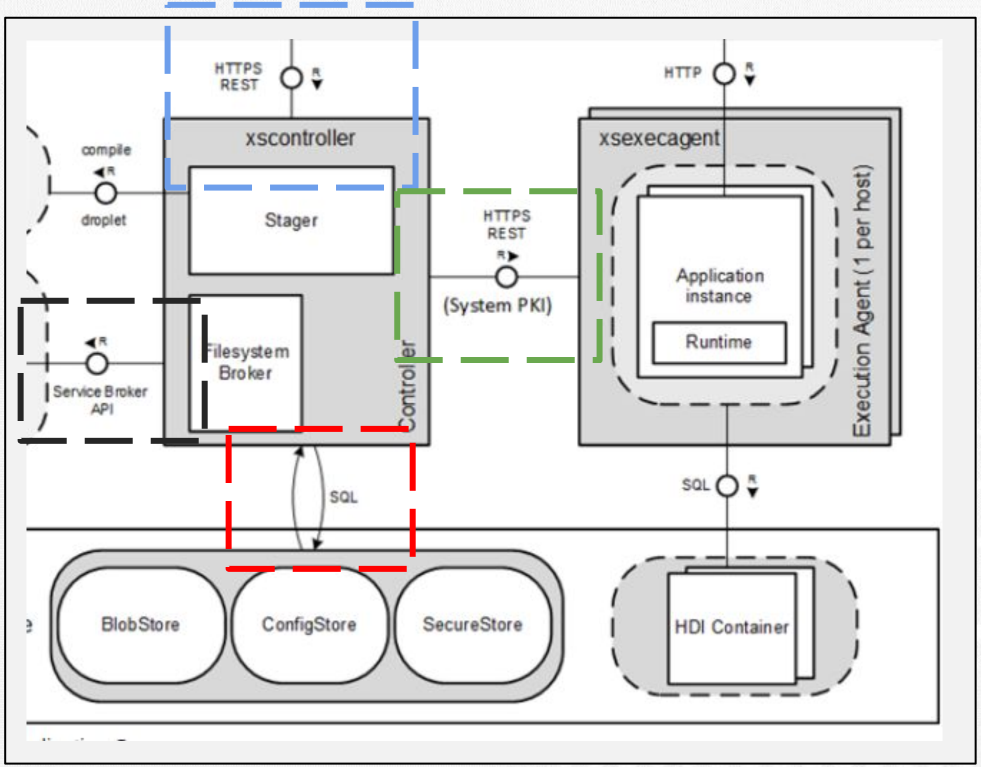
“The brain of the Cloud Foundry Environment”



- Manages spaces and organizations.

- Provides several REST API’s to access and maintain the whole environment.

Who is the Controller?



- API's exposed
- Internal connections to exec agents
- Connection to DB
- Service Brokers

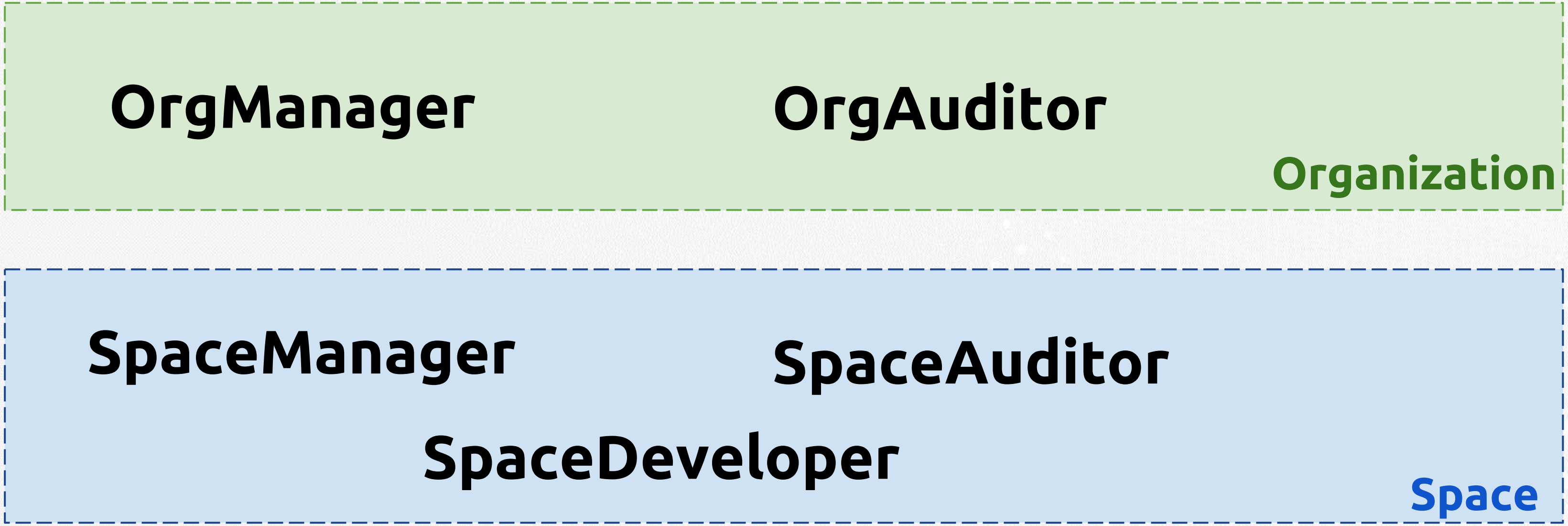
API’s exposed

- How to interact with them?
  - xs client
- Controller is shipped as **jar** file
- REST API’s Implemented using JAX-RS.
- Developed a parser:
  - Identifies all data for each web service .
  - Uniform output for further processing.
  - **Will go public at our github!**

Protections in place

- Organization isolation
    - ○ Administrative Level
- Space isolation
    - ○ Development Level
    - ○ Administrative Level
    - ○ OS Level
- Authorizations per Space & Organization
- Application isolation
    - ○ Bindings to services (i.e: DB)
- ...

Controller’s Authorization Model



OrgManager

OrgAuditor

Organization

SpaceManager

SpaceAuditor

SpaceDeveloper

Space

“**Auditors** can view space/organization resources, **excluding credentials**”

DEMO

onapsis

What happened?

- Spac … ivileged user … ccess to :
  - ke … tials

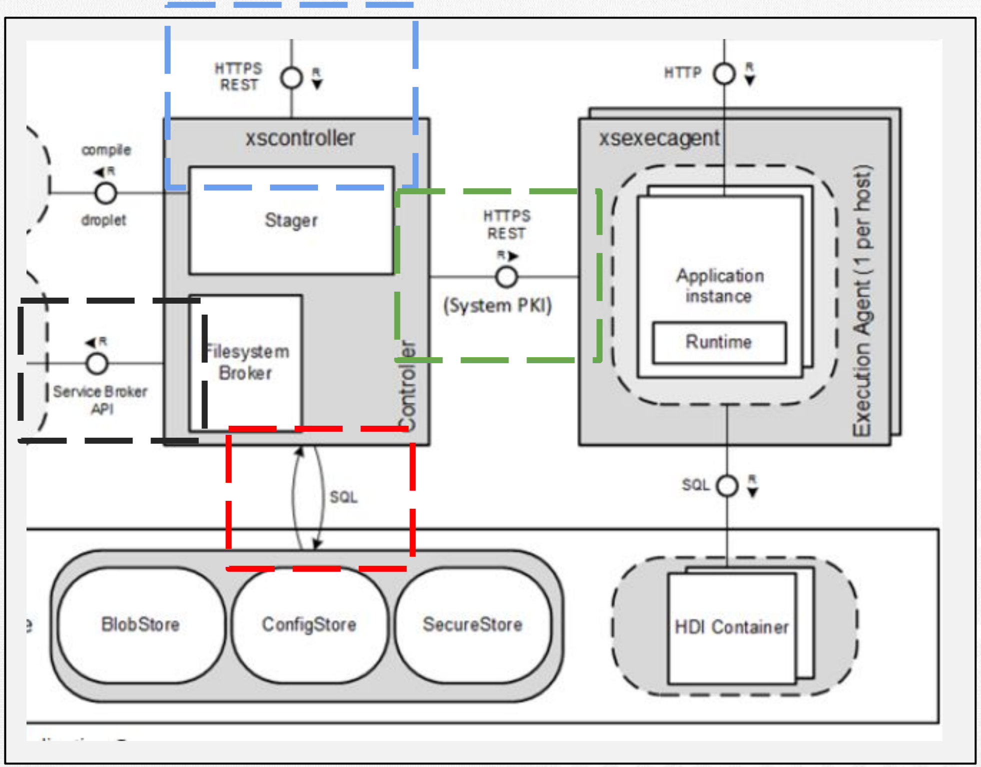
    ps bind … dentials

Broke o … he protections … d:

Authorizations per Spaces

Patched in **SAP Security Note 2589129**
CVE-2018-2375

Who is the Controller?



- **API's exposed**
- **Internal connections to exec agents**
- **Connection to DB**
- **Service Brokers**

Connection to DB

- The Controller, UAA and applications need to interact with the DB.

- **SYS_XS_RUNTIME** is the owner at DB level of the schema of XSA
  - Catalog Read
  - Inifile admin
  - User Secure store (I,D,R)

Finding I

- Found endpoint which wasn't sanitizing the input of a SQL statement.
- But..
- Same input, multiple statements.

The only way we found was… calling him!

DEMO

What happened?

- **1** vulnerabl
  1 endp API
- Unaut icate ker → Read SQ bles
- Only **l access.** N possible wi e **secstore.**

Patched in **SAP Security Note 2589129**
CVE-2018-2373

CVSS (v3):
5.3

What about the mentioned protections?

At the level of the Controller
most of them ….

Can be circumvented!

Although business data remains safe,
some technical data…

Post exploitation

Read controller's schema (SYS_XS_RUNTIME)

- Organizations, spaces, applications, services.
- XSA users info (authorizations)
- Source code of applications
- Environment data of apps
- ...

Read SYS schema

- System logs / users / configurations
- Information about DB (OS, hosts, tenants)
- ...

Post exploitation

Read controller's schema (SYS_XS_RUNTIME)

- Organizations, spaces, applications, services.
- XSA users info (authorizations)
- Source code of applications
- **Environment data of apps**
- ...

Read SYS schema

- System logs / users / configurations
- Information about DB (OS, hosts, tenants)
- ...

Environment data of apps

- Environment data about each app.
  - space
  - id’s (droplet, instance, app, etc)
  - services bound
  - more...
- **vcap_services** is not there, but..
- Passwords in plain text!
  - **service brokers running as apps** (users and passwords)
  - More passwords

Service broker open API

OPEN SERVICE BROKER API™

CREATE

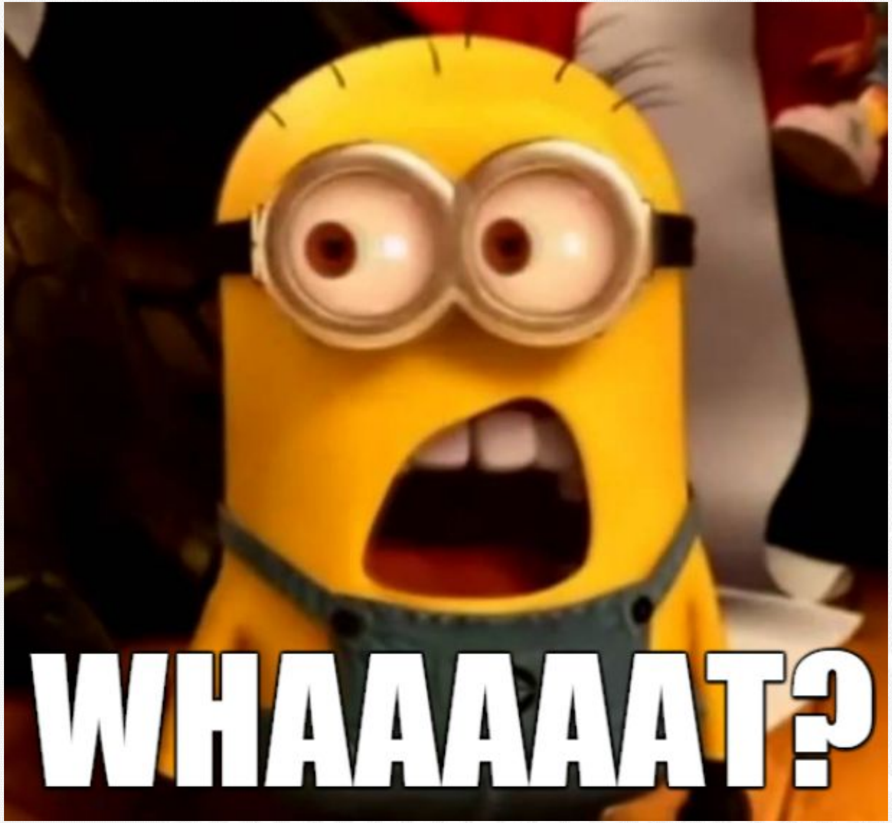
BIND

UNBIND

MODIFY

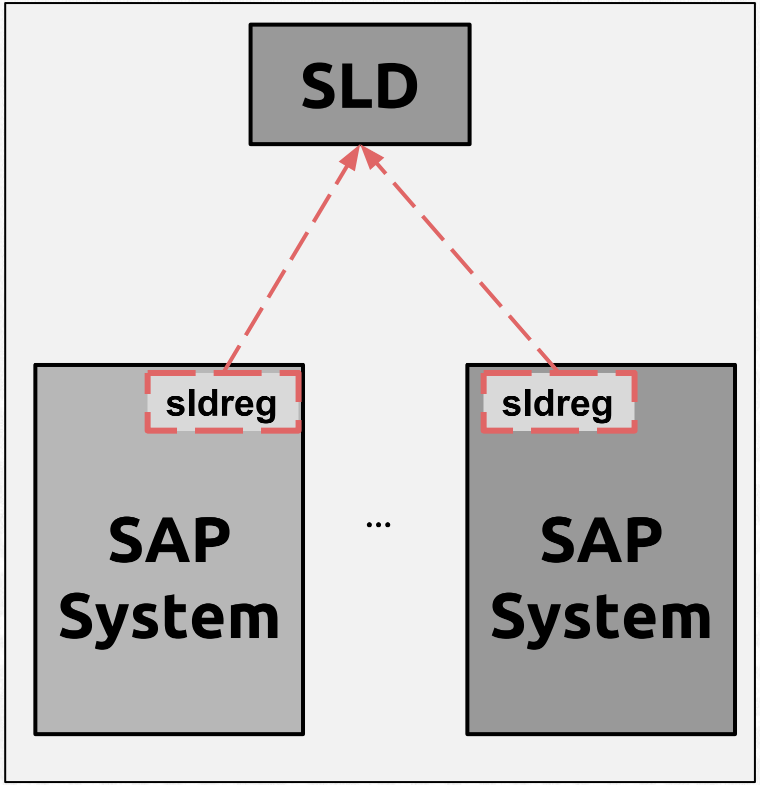
DELETE

but… that’s not all



Keep going!

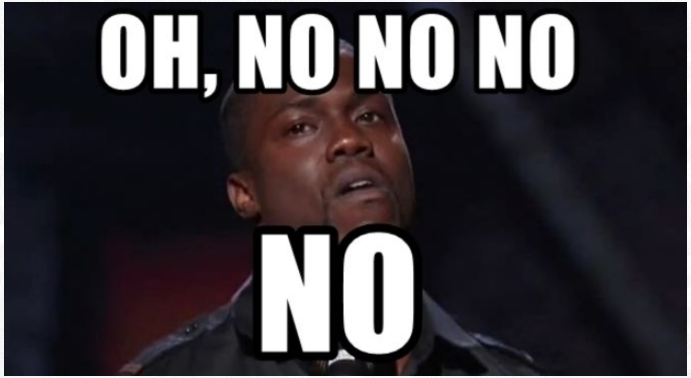
- SAP Landscape Directory? (SLD)
- XSA exposes an **authenticated** way of sending data to the SLD.
- Only **SpaceDeveloper's** of the **SAP** space can access them.



so ...?

BUT ....

- Leveraging the SQLi the credentials can be read
- The attacker can control the data sent to the SLD…
- Data is sent using XML…
- External Entities were possible



NO GOD PLEASE
NOOOOOOOOO

Wait! Is not so simple as it sounds!

- Data is sent to the **SLD**! Not back to the user!
  and…
- XML lib doesn't implemented **http://** protocol!

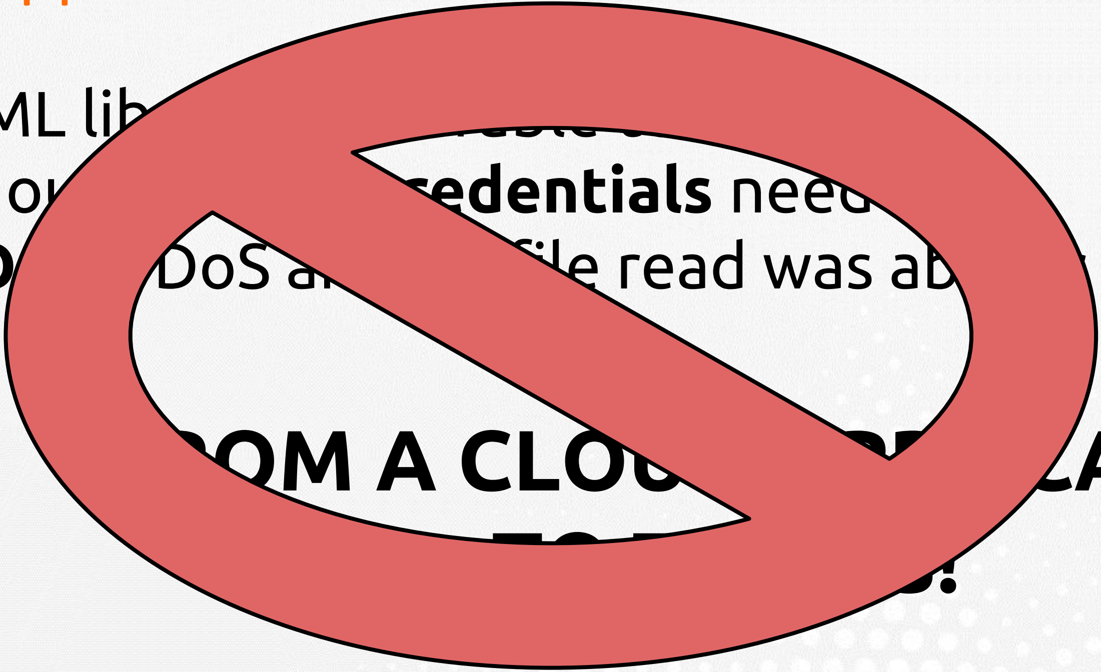
- Again, we had to call..

DEMO

onapsis

What happened?

- XML lib
- In o redentials need
- “O DoS a file read was ab SIDadm user.



OM A CLOU ATION

S!

Patched in **SAP SSN’s 2729710 & 2764283**
CVE-2019-0277 , CVE-2019-0265

What about protections?

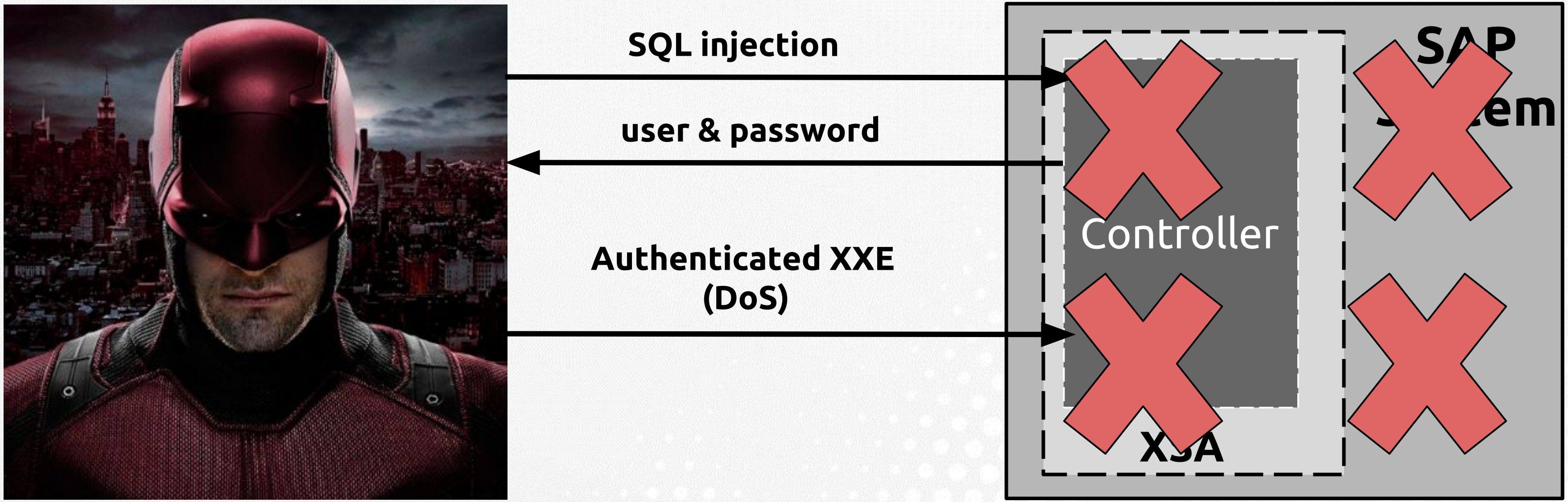
- Important files can be retrieved:
  - UAA files
  - Tomcat logs
  - Config files
  - ...

- **However, space isolation at OS level, worked very well.**

Recapping

onapsis

Full attack illustration v1



SQL injection

user & password

Authenticated XXE
(DoS)

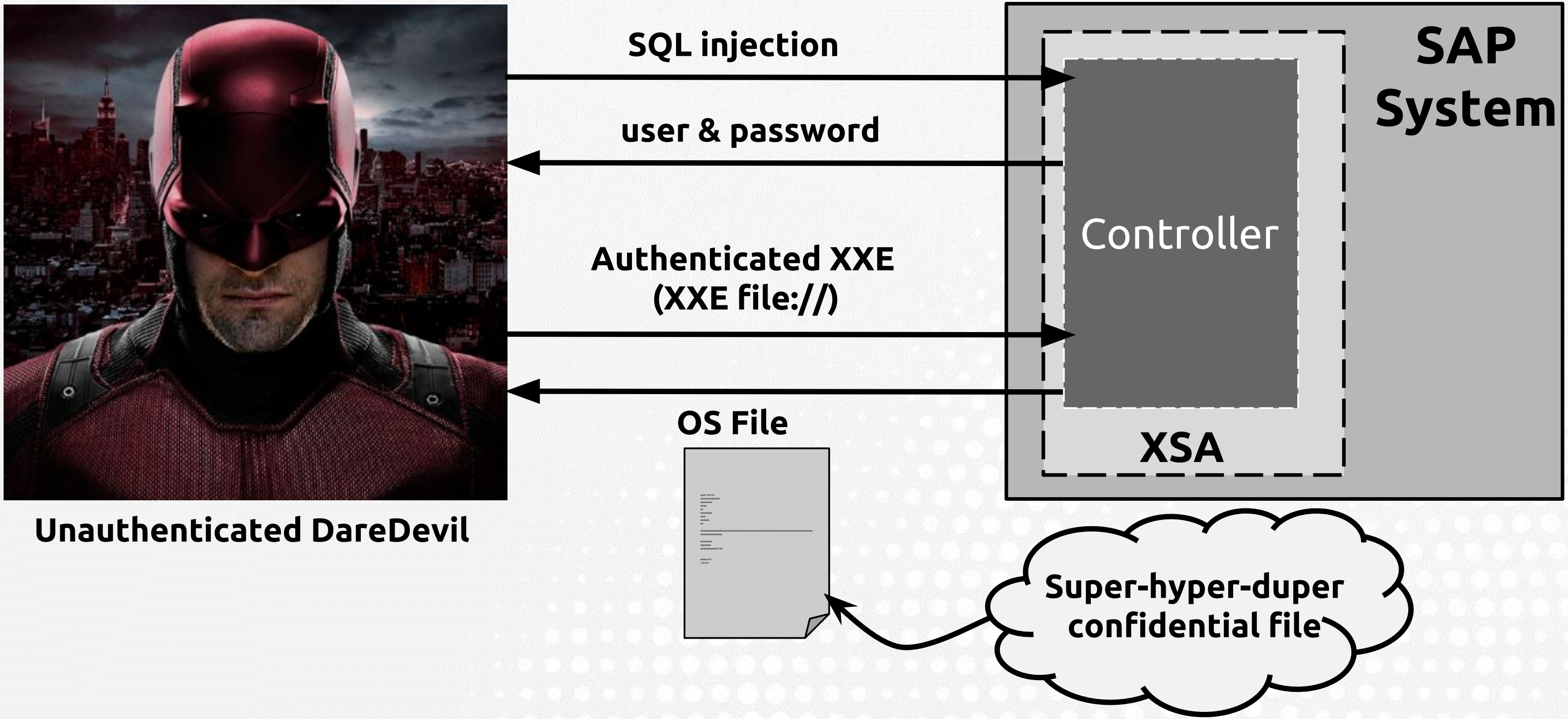
SAP
System

Controller

XSA

Unauthenticated DareDevil

Full attack illustration v2



SQL injection

user & password

Authenticated XXE
(XXE file://)

OS File

SAP System

Controller

XSA

Unauthenticated DareDevil

Super-hyper-duper confidential file

Conclusions

Takeaway

“In the land of the blind,
the one-eyed man is king”

Conclusions

- Cloud is **COMPLEX** and, therefore, **DIFFICULT.**
- A little hole can open a big gap.
- Sometimes, CVSS can obscure impact.

Conclusions

- Our research found other vulnerabilities too
  - User enumerations
  - Information disclosures
  - More ...
- Future/ongoing research…
  - Other components
  - Connections between components
  - Applications implementation

Questions?

onapsis

Thank you!

@onapsis @nahueldsanchez_ @partu18

info@onapsis.com

www.onapsis.com

onapsis