Hacking the Cloud

Gerald Steere – Microsoft C+E Red Team (@Darkpawh)

Sean Metcalf – CTO Trimarc (@pyrotek3)

Gerald Steere - @darkpawh

10+ years experience as a penetration tester and red team operator

Member of C+E Red Team since 2014

Speaker at BlueHat and Bsides Seattle

Spends work days happily smashing atoms in Azure

Sean Metcalf - @pyrotek3

Founder Trimarc, a security company.

Microsoft Certified Master (MCM) Directory Services

Speaker: Black Hat, BSides, DEF CON, DerbyCon, Shakacon, Sp4rkCon

Security Consultant / Security Researcher

Own & Operate ADSecurity.org
(Microsoft platform security info)

+

Cloud FTW!

What's in it for me?

Staying clean while being mean

Buzzword bingo with cloud lingo

Pathfinding, recon, and targeting in multiple dimension

Currency exchange – what do I do with all these hashes?

Happy fun exploit time (with demos)

Countermeasures and proper protection

What's in it for me?

What's in it for me?

Cloud matters for business

Your client probably uses it, whether you (or they) realize it or not

Many traditional techniques do not work

Same concepts but new ways of thinking

Hacker Quest

When we last saw our intrepid red team

Hired to red team SithCo

Have domain admin on a subsidiary domain

SithCo uses public cloud resources to host web applications

How do we leverage access to get into SithCo corporate?

Staying clean while being mean

Cause pissing off The Net is bad for business

Can I really go after my client's cloud deployments?

We are not lawyers.

If you're a professional you need one of those to talk to ALWAYS.

Lawful Evil is a perfectly valid alignment

Scope & Access will be more limited

Spell out enforced limitations in your reporting

Cloud providers typically require an approval process be followed

Attacking Azure, AWS, or Google Cloud Deployments

Requires preapproval by account owner (Azure and AWS)

Standard Rules of Engagement (RoE) stuff

Limited to customer owned resources

No DoS

Can include attempts to break isolation (Azure)

Buzzword Bingo

Do you have your card ready?

Accessibility modifiers

Public could

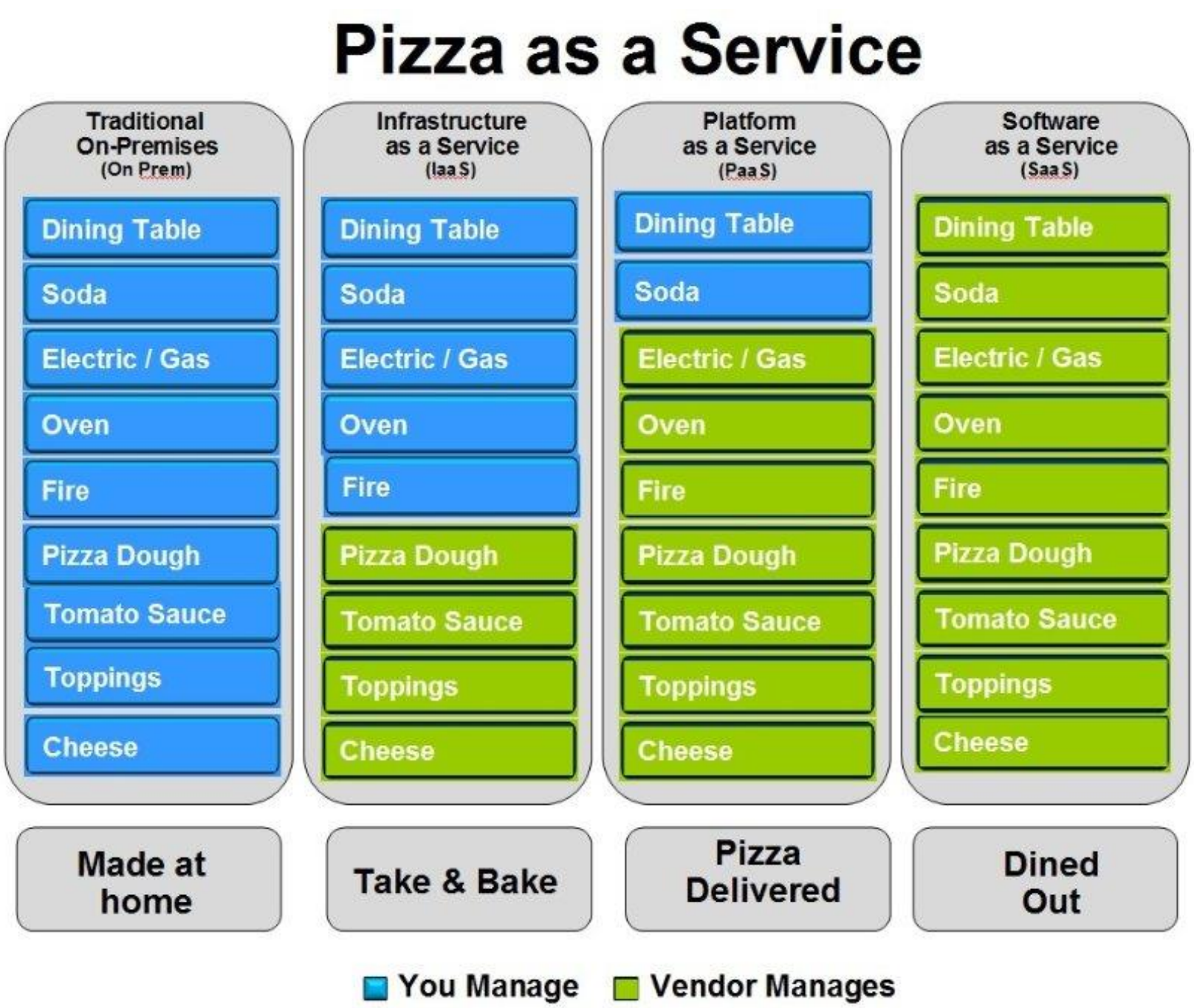
Private cloud

Hybrid cloud

https://www.stickermule.com/marketplace/3442-there-is-no-cloud

Buzzword Bingo

All the aaS

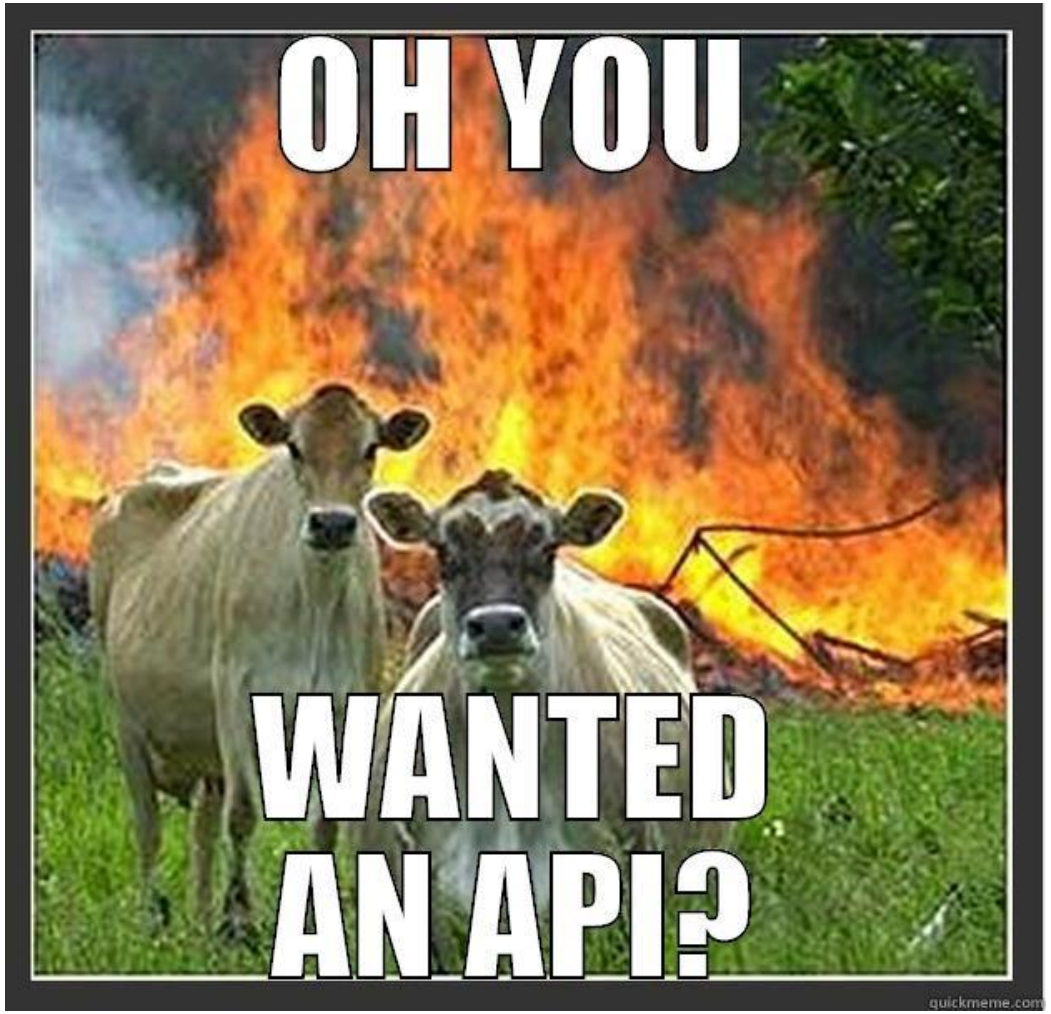


Albert Barron – https://www.linkedin.com/pulse/20140730172610-9679881-pizza-as-a-service

It’s not domain, but it’s still admin

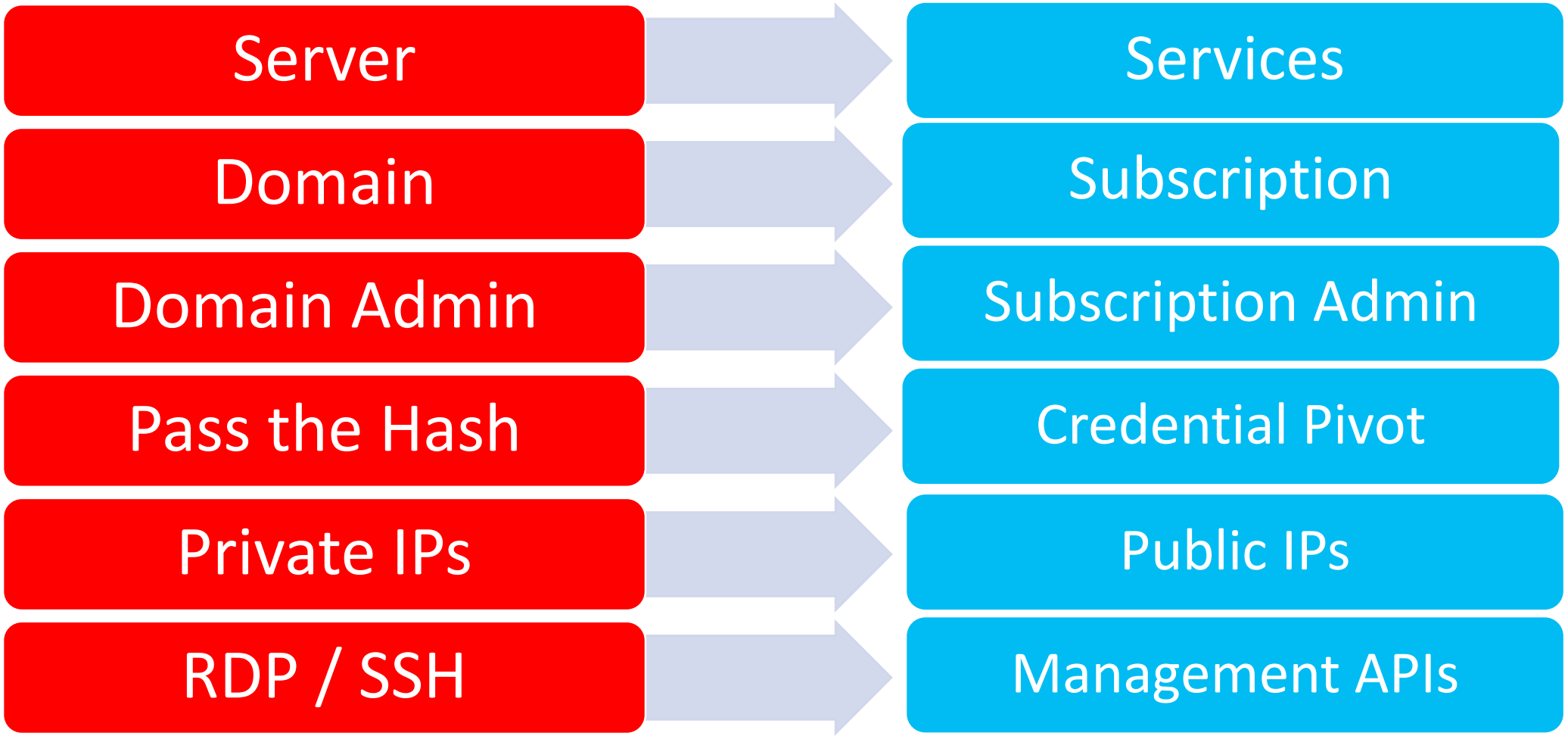
Cloud assets are managed under an account or subscription

Getting access to that layer is often equivalent to DA



Buzzword Bingo

CloudOS - Same ideas, different words



Faust and Johnson – Cloud Post Exploitation Techniques Infiltrate 2017 https://vimeo.com/214855977

Where's the data?

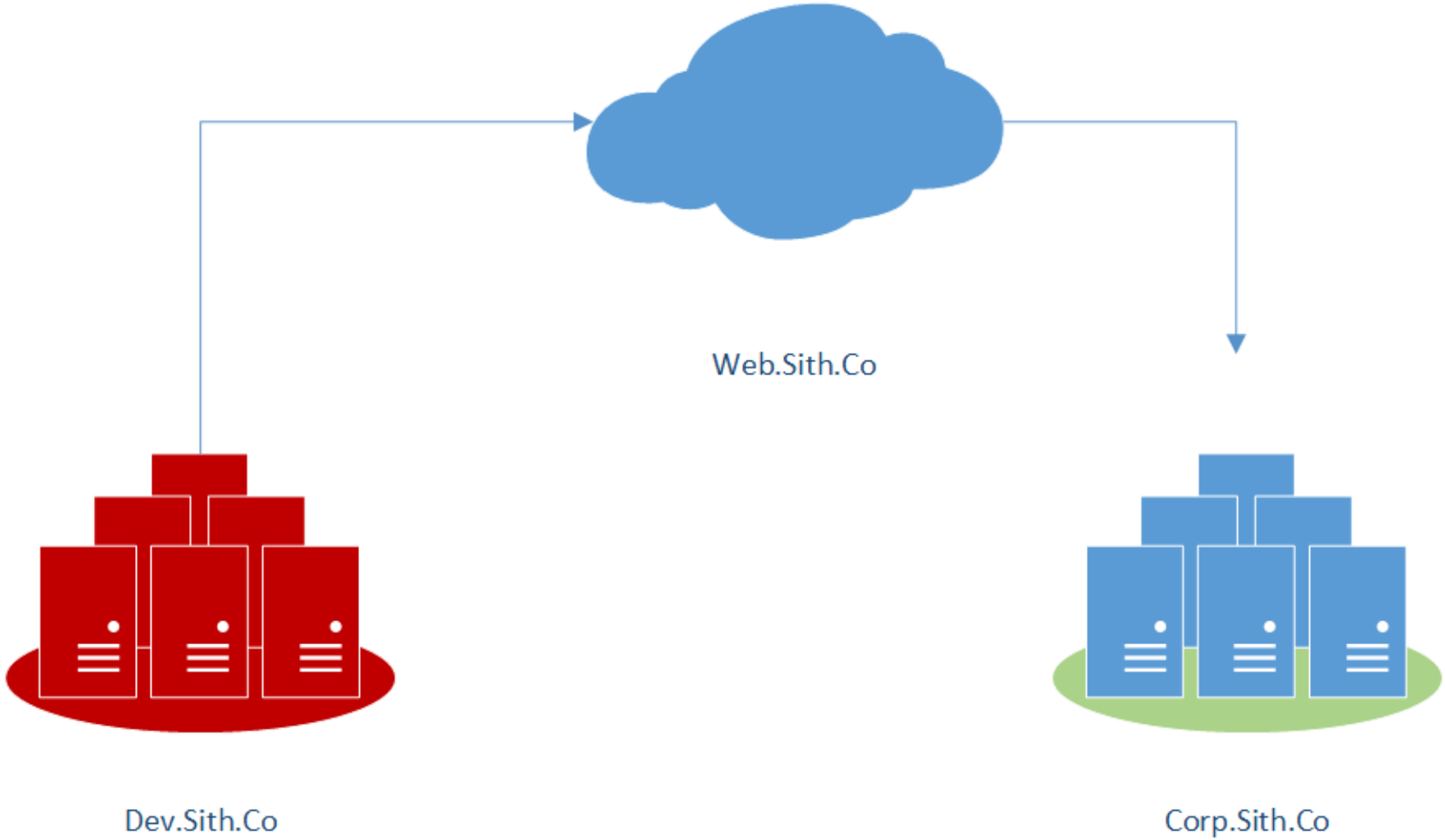
Cloud services rely on data storage for nearly everything

How is data stored in the cloud?

Do I need to attack the service or is the data my real goal?

Image: ©MITRE

SithCo's app hosting



Pathfinding, recon, and targeting in multiple dimension

How do I figure out I even need to look at the cloud?

Identifying Cloud Deployments

In the public cloud –

DNS is your best friend

Cloud Recon: DNS MX Records

- Microsoft Office 365:
  DOMAIN-COM.mail.protection.outlook.com
- Google Apps (G Suite):
  *.google OR *.googlemail.com
- Proofpoint (pphosted)
- Cisco Email Security (iphmx)
- Cyren (ctmail)
- GoDaddy (secureserver)
- CSC (cscdns)

Name	Value
outlook.com	116
pphosted.com	110
messagelabs.com	46
iphmx.com	34
ctmail.com	29
secureserver.net	25
cscdns.net	18
mimecast.com	18
google.com	15
m1bp.com	6
mb5p.com	6
googlemail.com	6
barracudanetworks.com	6

Cloud Recon: DNS TXT Records

MS = Microsoft Office 365

Google-Site-Verification = G Suite

Amazonses = Amazon Simple Email

OSIAGENTREGURL = Symantec MDM

AzureWebsites = Microsoft Azure

Paychex = Paychex financial services

Docusign = Docusign digital signatures

Atlassian-* = Atlassian services

Name	Value
MS	535
google-site-verification	242
adobe-idp-site-verification	86
docusign	80
v	54
globalsign-domain-verification	47
amazonses	31
atlassian-domain-verification	16
cisco-ci-domain-verification	11
dropbox-domain-verification	9
yandex-verification	6
OSIAGENTREGURL	6
bugcrowd-verification	4
cisco-site-verification	4
ios-enroll	3
have-i-been-pwned-verification	3
azurewebsites	3
android-mdm-enroll	2
status-page-domain-verifica...	2
android-enroll	2
paychex	1
Type	1
OLDMS	1
domain-verification	1
archiva-site-verification	1

Cloud Recon: SPF Records

SalesForce (salesforce.com, pardot.com, & exacttarget.com)

MailChimp (mcsv.net)

Mandrill (MailChimp paid app)

Q4Press (document collaboration)

Zendesk (support ticket)

Oracle Marketing (Eloqua.com)

Constant Contact (email marketing)

Postmark (mtasv.net)

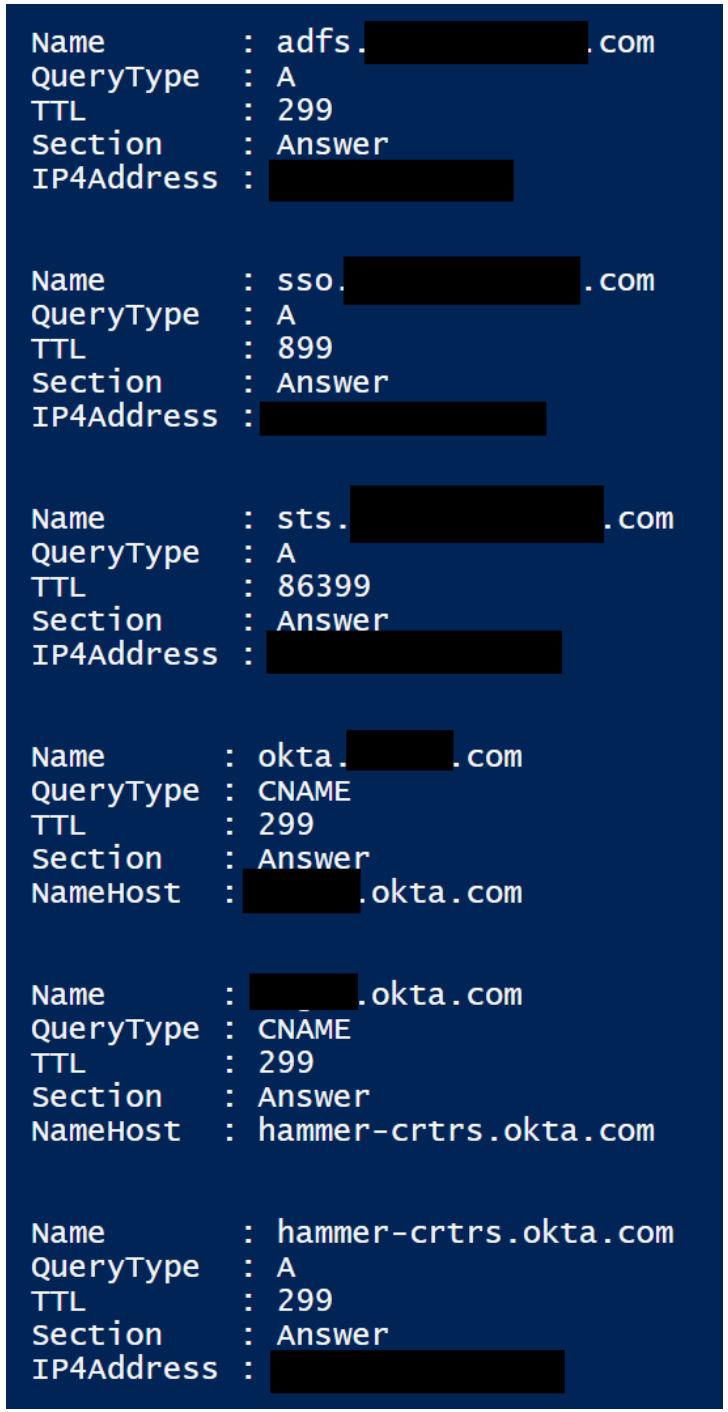
Name	Value
protection.outlook	180
pphosted.com	71
messagelabs.com	41
google.com	30
salesforce.com	30
mandrillapp.com	19
mcsv.net	19
pardot.com	17
q4press.com	16
exacttarget.com	12
mimecast.com	9
zendesk.com	8
oracle.com	8
eloqua.com	7
boardbooks.com	6
spf.messagelabs	6
qualtrics.com	5
clearslide.com	5
clickdimensions.com	5
constantcontact.com	4
satmetrix.com	4
microsoft.com	4
amazon.com	4

Discover Federation Servers

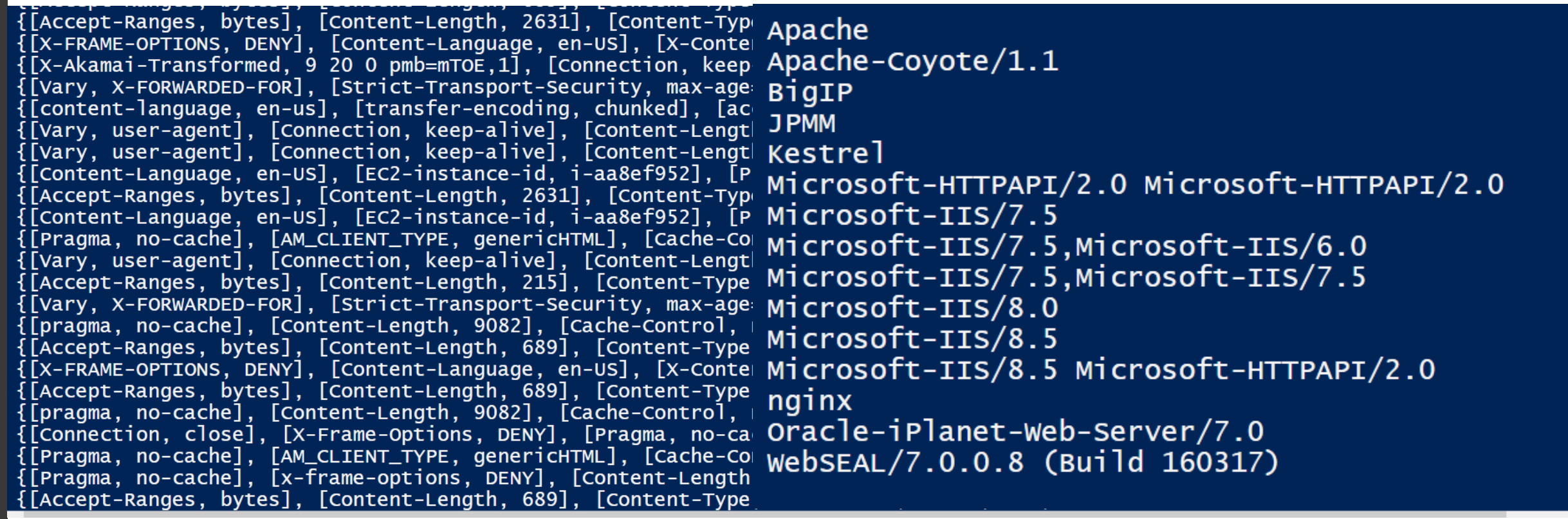
No standard naming for FS.

DNS query for:

- adfs
- auth
- fs
- okta
- ping
- sso



Federation Web Page Detail



TiPMix=0.505320029568542; path=/; Domain=okta.[redacted]ARR

L; expires=Wed, 11-Oct-2017 17:06:46 GMT; Max-Age=7776000; path=/; domain=.

OWA Version Discovery

Check for autodiscover subdomain (autodiscover.domain.com)

Connect to autodiscover web page (https://autodiscover.domain.com)

Copyright date effectively provides Exchange version:
2006 = Microsoft Exchange 2007

```
.tnarrow .officeFooter
{
    display: none;
}
</style>
<script>
//  flogon.js
//
//  This file contains the script used by Logon.aspx
//
//Copyright (c) 2003-2006 Microsoft Corporation  All rights r

/// <summary>
/// OnLoad handler for logon page
/// </summary>
window.onload = function ()
{
    // If we are replacing the current window with the logon p
    //
```

Cloud and Federation

Attackers go after Identity since that provides access to resources.

Modern auth

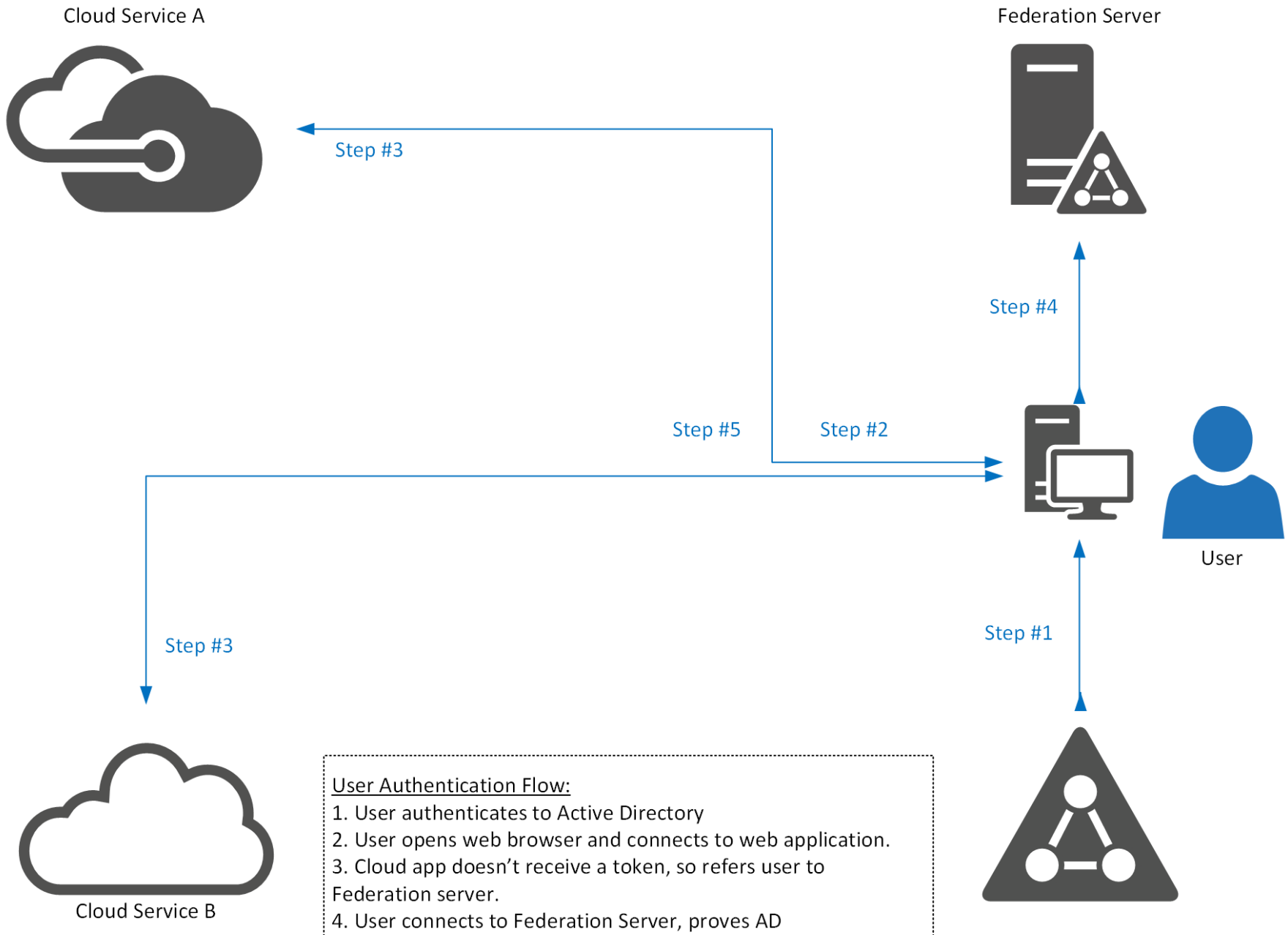
Cloud authentication and authorization is typically independent from the on-premises domain, though Federation may provide a path…

How you authenticate will depend on the specific cloud provider

More Buzzword Bingo:

- OAUTH
- OpenID
- SAML
- WS-Federation
- WS-Trust

Identity



User Authentication Flow:

1. User authenticates to Active Directory
2. User opens web browser and connects to web application.
3. Cloud app doesn't receive a token, so refers user to Federation server.
4. User connects to Federation Server, proves AD authentication, and receives token.
5. Connects back to cloud app providing token. User is allowed access based on data in token.

ADFS Federation Server Config

Federation server typically lives on the internal network with a proxy server in the DMZ.

Certificates installed on Federation server

- Service communication
- Token-decrypting
- Token-signing

Relying party trusts: cloud services and applications

Claim rules: determine what type of access and from where access is allowed.

Federation Key Points

Federation: trust between organizations leveraging PKI (certificates matter)

Cloud SSO often leverages temporary or persistent browser cookies (cookies provide access)

Several protocols may be supported, though typically SAML. (protocols and versions matter)

Federation server (or proxy) is on public internet via port 443 (HTTPS).

How to steal identities – federated style

Federation is effectively Cloud Kerberos.

Own the Federation server, own organizational cloud services.

Token & Signing certificates ~= KRBTGT (think Golden Tickets)

Steal federation certificates to spoof access tokens (Mimikatz fun later).

On-Premises Cloud Components

How do we get those identities into the cloud anyways?

Active Directory & the Cloud

Active Directory provides Single Sign On (SSO) to cloud services.

Some directory sync tools synchronizes all users and their attributes to cloud service(s).

Most sync engines only require AD user rights to send user and group information to cloud service.

Most organizations aren’t aware of all cloud services active in their environment.

Express Permissions for Azure AD Connect

Permissions for the created AD DS account for express settings

The account created for reading and writing to AD DS have the following permissions when created by express settings:

Permission	Used for
• Replicate Directory Changes • Replicate Directory Changes All	Password sync
Read/Write all properties User	Import and Exchange hybrid
Read/Write all properties iNetOrgPerson	Import and Exchange hybrid
Read/Write all properties Group	Import and Exchange hybrid
Read/Write all properties Contact	Import and Exchange hybrid
Reset password	Preparation for enabling password writeback

https://docs.microsoft.com/en-us/azure/active-directory/connect/active-directory-aadconnect-accounts-permissions

Identity

Custom Permissions for Azure AD Connect

Feature	Permissions
msDS-ConsistencyGuid feature	Write permissions to the msDS-ConsistencyGuid attribute documented in Design Concepts - Using msDS-ConsistencyGuid as sourceAnchor.
Password sync	• Replicate Directory Changes • Replicate Directory Changes All
Exchange hybrid deployment	Write permissions to the attributes documented in Exchange hybrid writeback for users, groups, and contacts.
Exchange Mail Public Folder	Read permissions to the attributes documented in Exchange Mail Public Folder for public folders.
Password writeback	Write permissions to the attributes documented in Getting started with password management for users.
Device writeback	Permissions granted with a PowerShell script as described in device writeback.
Group writeback	Read, Create, Update, and Delete group objects in the OU where the distributions groups should be located.

https://docs.microsoft.com/en-us/azure/active-directory/connect/active-directory-aadconnect-accounts-permissions

Currency exchange – what do I do with all these hashes?

I never liked buying tokens, but that's all these things take

Spending our horde

I’ve got all these hashes and no where to go

No matter how many times you’ve popped the KRBTGT account, your cloud provider really doesn’t care

Creds, creds never change

Certificates, certificates, certificates!

Popping dev boxes has never been more productive

You do know mimikatz can also export certificates, right?

```
mimikatz # crypto::certificates /systemstore:local_machine /store:my /export
 * System Store  : 'local_machine' (0x00020000)
 * Store         : 'my'

 0. example.domain.local
        Key Container  : example.domain.local
        Provider       : Microsoft Software Key Storage Provider
        Type           : CNG Key (0xffffffff)
        Exportable key : NO
        Key size       : 2048
        Public export  : OK - 'local_machine_my_0_example.domain.local.der'
        Private export : OK - 'local_machine_my_0_example.domain.local.pfx'
```

What is old is new again

Password Spraying involves attempting authentication with a single password against all users before moving on to the next password.

Works against Cloud services: email, IM, etc.

Low & Slow: 1 to 2 per hour

Often works against VPN as well.

Password spraying tools

OWA-Toolkit: https://github.com/johnnyDEP/OWA-Toolkit

MailSniper: Invoke-PasswordSprayOWA
https://github.com/dafthack/MailSniper

Patator: https://github.com/lanjelot/patator

LyncSniper: https://github.com/mdsecresearch/LyncSniper
https://www.mdsec.co.uk/2017/04/penetration-testing-skype-for-business-exploiting-the-missing-lync/

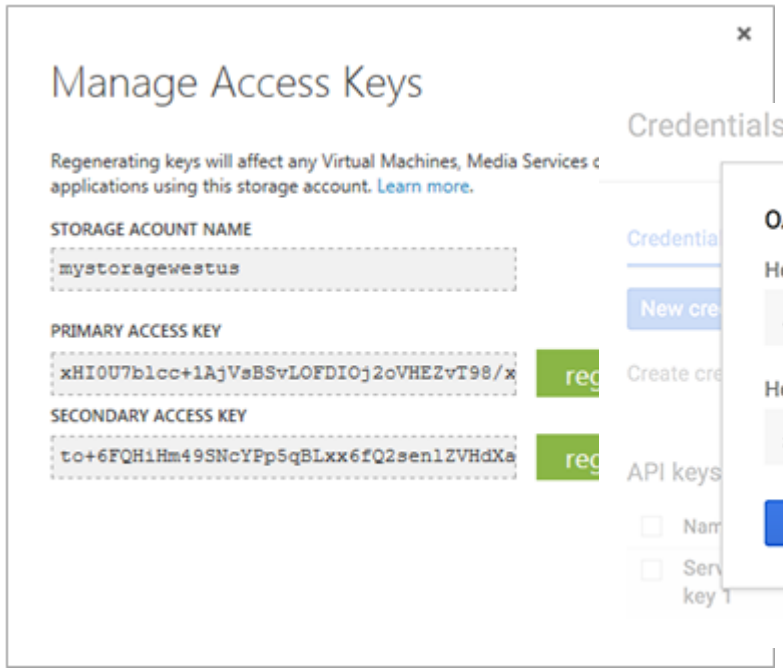
The authors have not evaluated these tools. Always test before use.

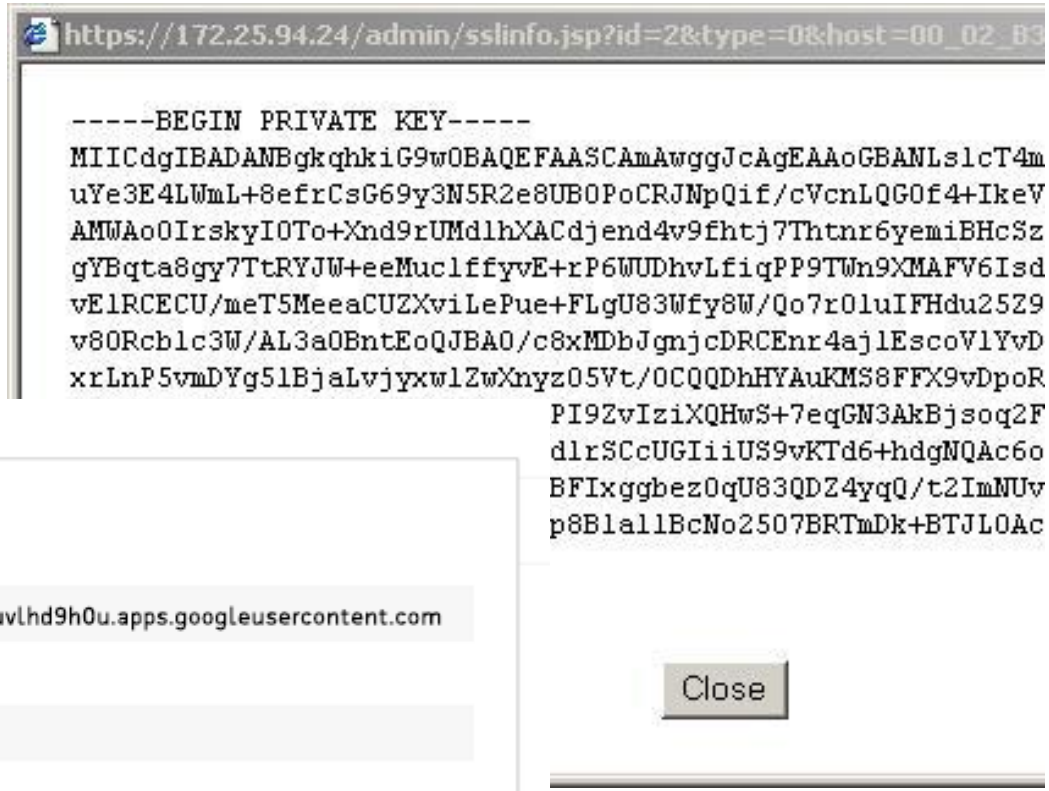
Currency exchange

DevOops

DevOps probably has what you are looking for

- API keys and shared secrets for the win
- Source code access for fun and profit
- How are these deployments done anyways?





Currency exchange

Where Are API Keys? GitHub!

accessKeyId and secretAccessKey are leaking #9

Open **jingidy** opened this issue on May 30, 2013 · 0 comments

jingidy commented on May 30, 2013

While running mocha tests for my project, two global leaks were detected due to the amazon-ses module.

Please see test case here:
https://gist.github.com/jingidy/5682149

https://github.com/jjenkins/node-amazon-ses/issues/9

```
1  var ApiBuilder = require('claudia-api-builder'),
2      api = new ApiBuilder();
3
4  module.exports = api;
5
6  AWS.config.update({
7    "accessKeyId": "AKIA[redacted]",
8    "secretAccessKey": "[redacted]",
9  }
10 );
```

https://hackernoon.com/how-to-use-environment-variables-keep-your-secret-keys-safe-secure-8b1a7877d69c

The circle of access

Access between on-premises and cloud deployments often a two way street

On-premises -> cloud typically involves identifying credentials

Is there a way back?

Are there shared authentication methods?

The circle of access

What is the likelihood this cloud service needs to access resources from on-premises?

Happy fun exploit time

Pray to the demo gods, pray I say!

Demo stuff here

There should be a fun live demo here if everything goes right

Countermeasures and proper protection

Closing my eyes and hoping it goes away isn't going to work, is it?

Giving useful advice

Telling your client to close up shop and moving back into the basement is probably a non-starter

Clouds do provide real business benefits and can improve security when done right

How can the “cloud” be secured?

Giving useful advice: The Basics

- Properly handle, store, and mange credentials and secrets
  - You aren’t storing those access keys in GIT are you?
  - Clouds do provide managed secret stores
  - Make it easy for DevOps to do the right thing
- Enforce MFA on all accounts
  - If it can’t have MFA, limit it as much as possible and monitor it

Giving useful advice: Securing Federation

Protect Federation servers at the same level as Domain Controllers.

Use a proxy server to limit communication directly with federation server inside the network.

Audit cloud authentication by logging Federation auth events & send to SIEM.

Enable multifactor authentication for all admin accounts & preferably all cloud accounts.

Control Cloud authentication via Federation rules.
Example:

- Internal network access provides single sign-on
- External access requires username, password, and two-factor authentication

Giving useful advice

Many of the basics remain the same

- Least privilege is key and poorly understood in many cloud implementations
- Least access, use the security features provided by the cloud

Credential management is hard in a connected world – this is an massive opportunity for attackers

Monitoring and alerting

It’s not just for your network any more

Defenders need to work with DevOps to make sure that cloud resources and data are considered in defensive designs

Different cloud providers provide different tools for managing security

Defenders must be familiar with the tools from cloud providers used by their client

Log collection and management needs to include cloud assets

You do know what your assets are, right?

Assume breach!

Conclusion

Are we there yet?

References

Infiltrate 2017: Cloud Post Exploitation Techniques - Andrew Johnson & Sacha Faust
https://vimeo.com/214855977

Azure Security: https://portal.msrc.microsoft.com/en-us/engage/pentest

AWS Security: https://aws.amazon.com/security/penetration-testing/

Google Cloud Security: https://cloud.google.com/security/

MailSniper: https://github.com/dafthack/MailSniper

Patator: https://github.com/lanjelot/patator