

From Zero to Secure in 1 Minute

Securing IaaS

Nir Valtman & Moshe Ferber

The NCR logo consists of a green square containing a white circular icon with a stylized 'N' and the letters 'NCR' to its right.

DEFCON #23

The CSAIL logo features the text 'CSAIL' in large blue letters, with 'cloud security alliance' in smaller orange letters to its right, and 'ISRAEL' in blue letters below 'CSAIL'.

About us

Moshe Ferber

Nir Valtman

- **Passionate about information security.**
- **Involved in numerous startups and initiatives** – sometimes with success 😊, sometimes not ☹️
- **Popular industry speakers and lecturers** – that's why we are here.

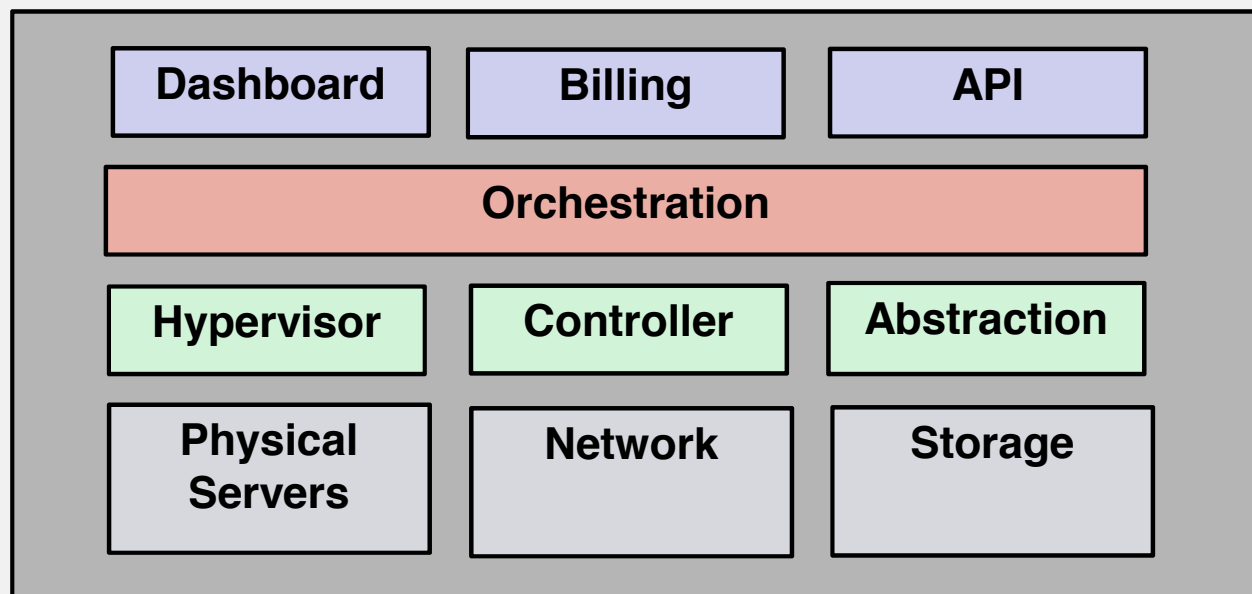
- **Instructor for Cloud Security (CCSK)** – that is what I really like doing.

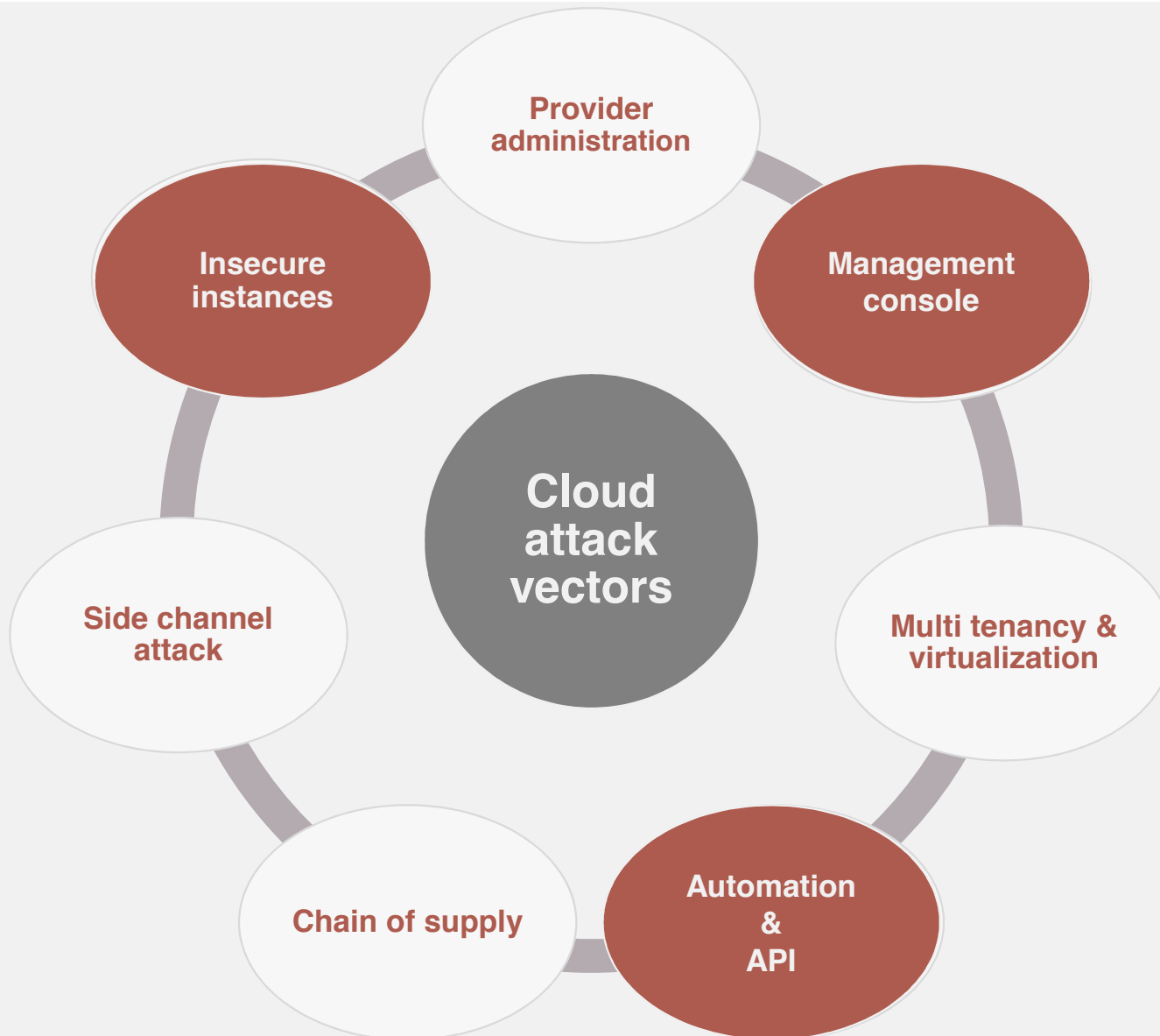
- **CISO Retail in NCR Corporation.**
 - We own a private cloud & offering SaaS
 - Yes we do security!

About the talk

Cloud security challenges and benefits

And more specifically, using IaaS automation and orchestration features for increasing security on our servers.





Anatomy of a cloud hack – BrowserStack story

Shell shock
vulnerability
on unused
server

Found API
key on
hacked
server

Using API
key opened
a firewall
rule and
launch an
instance

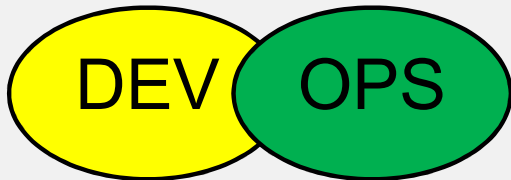
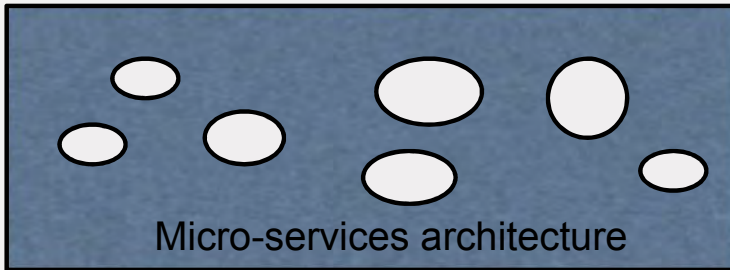
Attached a
backup
volume to
the instance

Found
database
credential
on backup
device

Connected
to DB

Source: <https://www.browserstack.com/attack-and-downtime-on-9-November>

Architecture & Deployments is changing



Continuous delivery

The billing cycle is reducing

Google slashes cloud platform prices ... again

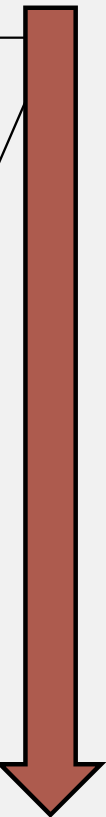
1 hour

Microsoft will offer Azure by the minute to take on Amazon's cloud

10 min

MICROSOFT FOLLOWS GOOGLE WITH BY-THE-MINUTE CLOUD BILLING

1 min



How to do security when servers alive for 10 minutes?

Patch
management

Maintenance
windows

Periodic
vulnerability
scanning

Hardening

Introducing



Launch

Configure
and harden

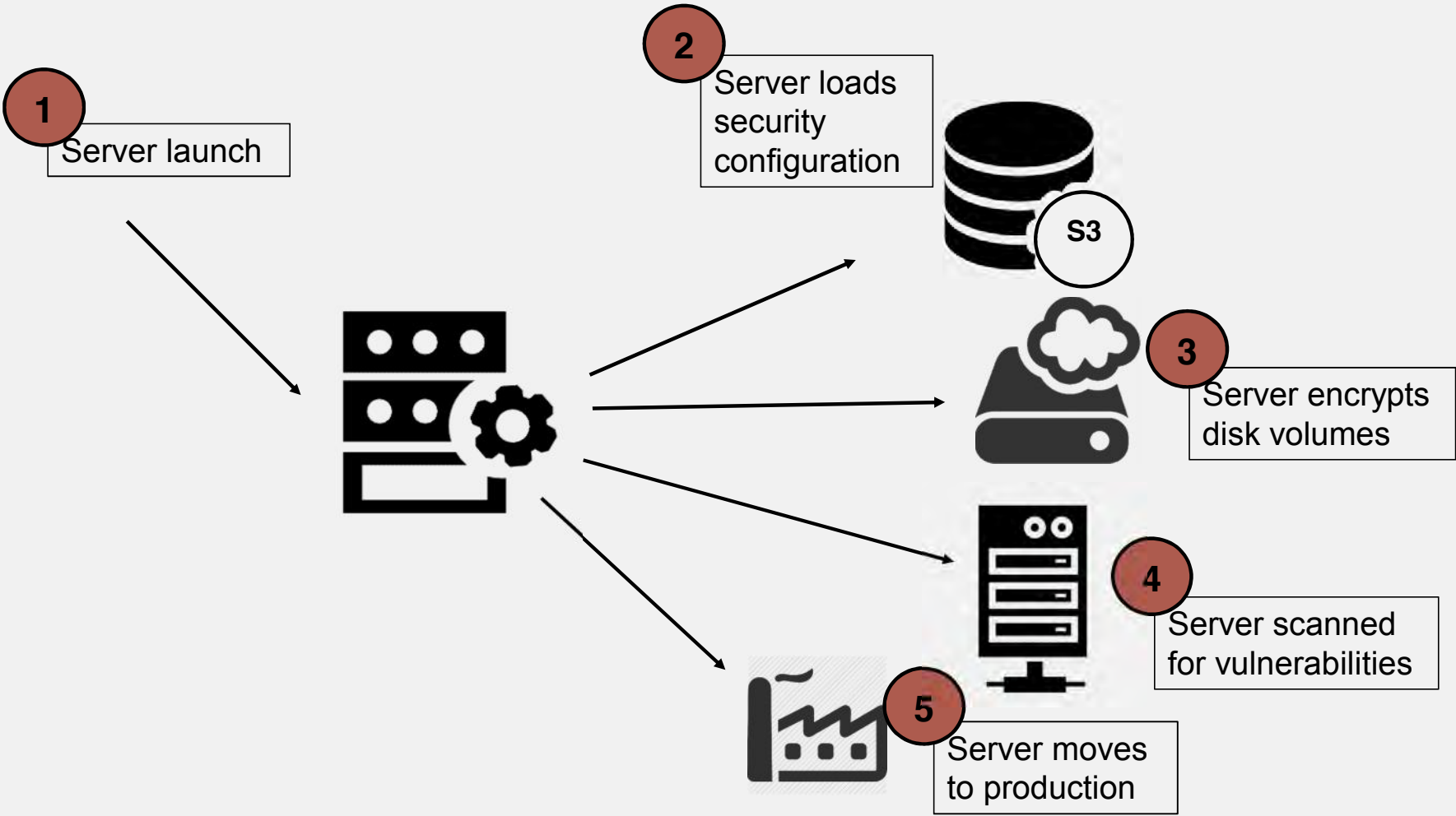
Scan

Move to
production

Source code: <http://www.cloudefigo.org/>

Based on the work made by Rich Mogull from Securosis
<https://github.com/rmogull/PragmaticNetSecManagement>

Cloudefigo Lifecycle



Components

Object Storage - AWS S3

A storage architecture that manage data as object. Files are stored along with metadata and unique identifier. Access is usually by HTTP/S.



Security Scanner

We use Nessus since it's very popular. There are commercial products with built in integration to AWS though:

<http://www.tenable.com/products/nessus>



CloudInit

CloudInit is a package (originally introduced by Ubuntu) that handles early initialization of a cloud instance.

Configuration management

We have used Chef, because it is open source and very integrative to our environment.

<https://www.getchef.com/>



AWS IAM Role

IAM roles provide permissions for resources. Instances can be assign with an IAM Roles that will determine which resources inside AWS the instance can access.

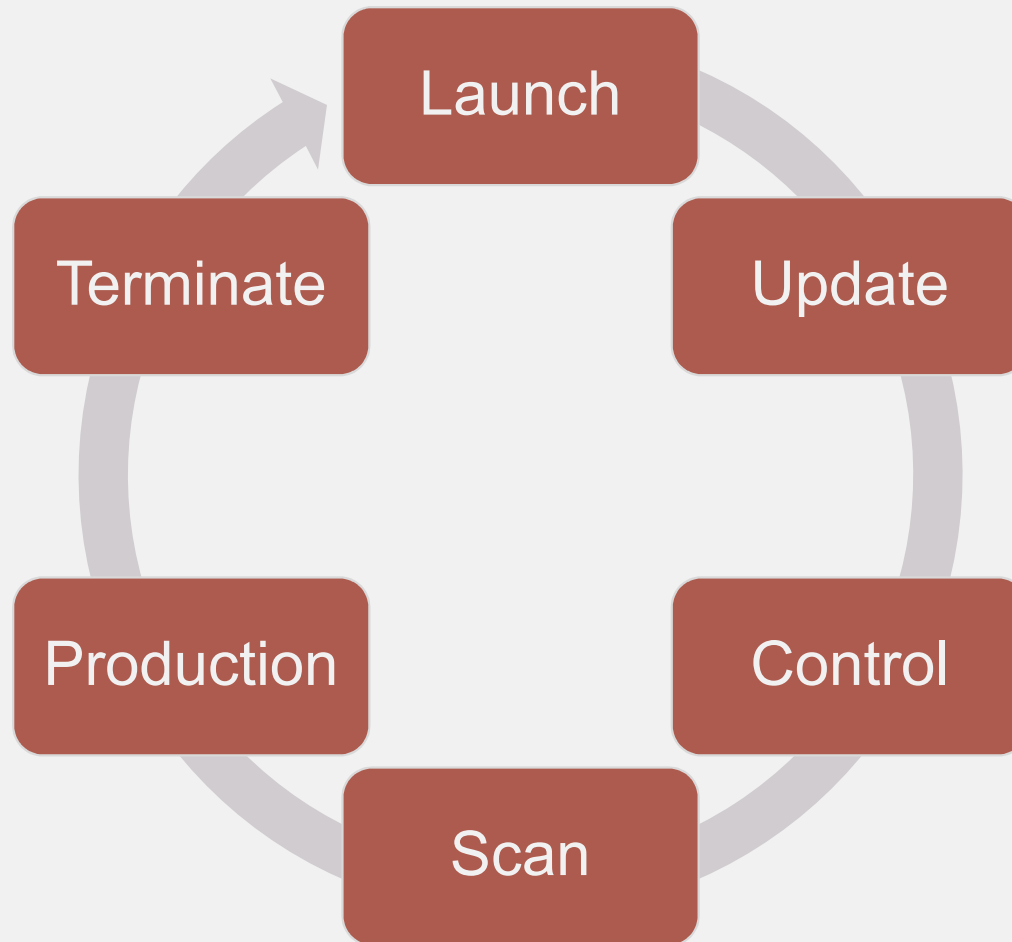
Encryption

We have used Full Disk Encryption open source software called "dm-crypt".

<https://code.google.com/p/cryptsetup/wiki/DMCrypt>



Instance Lifecycle



Launch



Prepare

CloudInit

- Each machine manage its own attributes
 - Encryption keys
 - Remediation vs production groups.
- Management of these attributes require permissions.
- Permissions during launch > production
- Thus, a dynamic IAM role is required.

Launch

Prepare

CloudInit

```
IAMBasicPolicy.config
File Path: ~/PycharmProjects/CloudInit/AWS/IAMBasicPolicy.config
IAMBasicPolicy.config
4   {
5     "Sid": "Stmt1413488885000",
6     "Effect": "Allow",
7     "Action": [
8       "s3:CreateBucket",
9       "s3>DeleteBucket",
10      "s3:GetObject",
11      "s3:ListBucket",
12      "s3:PutBucketPolicy",
13      "s3:PutObject"
14    ],
15    "Resource": [
16      "arn:aws:s3:::BUCKETNAME"
17    ]
18  },
19  {
20    "Sid": "Stmt1413489080000",
21    "Effect": "Allow",
22    "Action": [
23      "s3:GetObject",
24      "s3:ListBucket"
25    ],
26    "Resource": [
27      "arn:aws:s3:::config-cloudsec"
28    ]
29  },
30  {
31    "Sid": "Stmt1413548658000",
32    "Effect": "Allow",
33    "Action": [
34      "ec2:DescribeInstanceAttribute",
35      "ec2:DescribeInstances",
36      "ec2:DescribeSecurityGroups",
37      "ec2:ModifyInstanceAttribute",
38      "ec2:RunInstances",
39      "ec2:CreateTags"
40    ],
41    "Resource": [
42      "*"
43    ]
44  },
}
```

Line 2 Col 27 (none) Unicode (UTF-8) Unix (LF) 1,301 / 82 / 64

Launch

```
graph TD; Launch[Launch] --- Prepare[Prepare]; Launch --- CloudInit[CloudInit];
```

Prepare

CloudInit

- Executed in root permissions when image is launching.
- Responsible for building the infrastructure for the following steps.

Launch

Prepare

CloudInit



Update



**OS
update**

Pre-
requisites

- CloudInit to update & upgrade software packages.
- Primary goal is to make sure the cloud instance is secure once upgraded.

Update

OS
update

Pre-
requisites

- CloudInit to install the software packages required to operate:
 - Python + pip + wheel.
 - AWS SDK (Boto)
 - Chef Client + Chef SDK (PyChef)
- Download configurations and scripts from S3:
 - Cloudefigo script.
 - Chef client initialization files.
- Cloudinit to create and attach a volume for application files and data.

Update

OS
update

Pre-
requisites



Control

**Chef
Registration**

Encrypt

- The Chef clients register to the Chef Management server using the initialization files loaded from S3.
- Once the client is registered, a policy is loaded and enforced on the instance.

Control

Chef
Registration

Encrypt

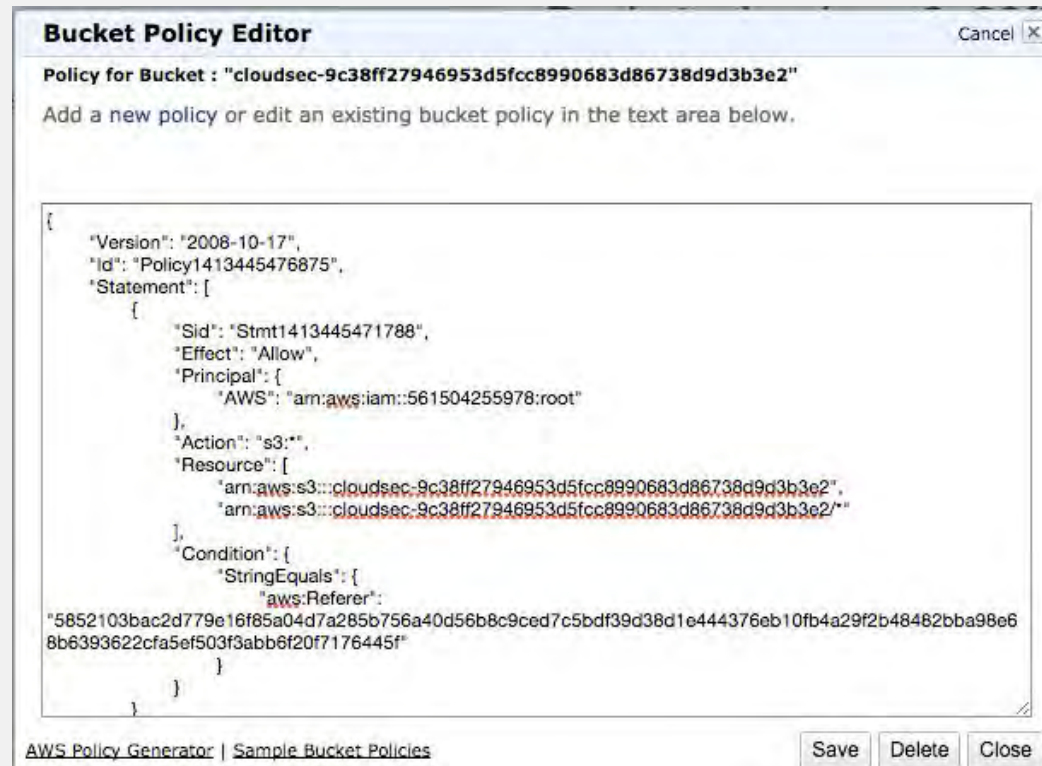
- The volume to be encrypted using randomly generated key.
 - The key is kept in S3 for later use.
- The application database to be installed in the encrypted volume.

Control

Chef
Registration

Encrypt

- Dynamic S3 policy: access to key require a referrer header that is generated based on attributes from the instance.



Bucket Policy Editor Cancel

Policy for Bucket : "cloudsec-9c38ff27946953d5fcc8990683d86738d9d3b3e2"

Add a new policy or edit an existing bucket policy in the text area below.

```
{
  "Version": "2008-10-17",
  "Id": "Policy1413445476875",
  "Statement": [
    {
      "Sid": "Stmt1413445471788",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::561504255978:root"
      },
      "Action": "s3:*",
      "Resource": [
        "arn:aws:s3:::cloudsec-9c38ff27946953d5fcc8990683d86738d9d3b3e2",
        "arn:aws:s3:::cloudsec-9c38ff27946953d5fcc8990683d86738d9d3b3e2/"
      ],
      "Condition": {
        "StringEquals": {
          "aws:Referer":
            "5852103bac2d779e16f85a04d7a285b756a40d56b8c9ced7c5bdf39d38d1e444376eb10fb4a29f2b46482bba98e68b6393622cfa5ef503f3abb6f20f7176445f"
        }
      }
    }
  ]
}
```

AWS Policy Generator | Sample Bucket Policies Save Delete Close

Control

Chef
Registration

Encrypt



Scan

```
graph TD; Scan[Scan] --- Automatic[Automatic Scan]; Scan --- Analyze[Analyze];
```

**Automatic
Scan**

Analyze

- A vulnerability scan to be launched automatically by CloudInit script.
- The deeper the scan, the longer it takes to move to production.

Scan

Automatic
Scan

Analyze

- The results of the scan are analyzed by the Cloudefigo script.
- Based on scan results – the instance to move to production or remain in the remediation group.
- The lowest security risk severity can be defined.

Scan

Automatic
Scan

Analyze



Production

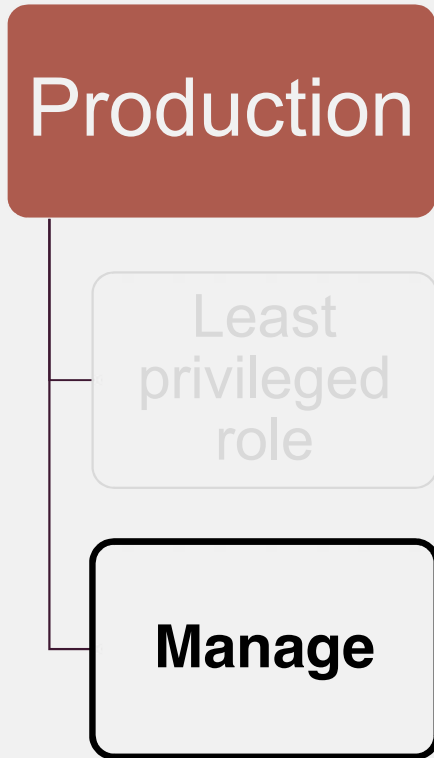
**Least
privileged
role**

Manage

- Reminder: Permissions in launch > production
- IAM role permissions reduced dynamically - contains read only access

```
IAMStrictPolicy.config
File Path: ~/PycharmProjects/SecureCloudNit/AWS/IAMStrictPolicy.config
IAMStrictPolicy.config
1  {
2    "Version": "2012-10-17",
3    "Statement": [
4      {
5        "Sid": "Stmt1413489080000",
6        "Effect": "Allow",
7        "Action": [
8          "s3:GetObject",
9          "s3:ListBucket"
10       ],
11       "Resource": [
12         "arn:aws:s3:::BUCKETNAME"
13       ]
14     }
15   ]
16 }
17
```

Line 12 Col 34 | (none) | Unicode (UTF-8) | Unix (LF) | Last saved: 10/23/14, 12:29:49 AM | 260 / 19 / 17



- For the ongoing operations – a compensating control is required to locate unmanaged instances.
- Cloudefigo management script lists cloud instances and validates they are managed by Chef.
- Unmanaged instance can move to remediation, forensics (not implemented in the current version)

Production

Least
privileged
role

Manage



Terminate

instance

Encryption
Keys

- The life cycle ends once a server is terminated along with:
 - Attached volumes
 - IAM role

Terminate

instance

**Encryption
Keys**

- The instance data still exist in backups/snapshots or provider storage.
- Encryption keys to be deleted with instance in order to make sure the backup data remain inaccessible (not implemented in this version)

Questions



Wrapping up

The new software architecture and applications delivery in cloud module disrupts traditional correctives controls

We need to adopt new thinking to automate security.

Think how security automation can help you in moving your infrastructure forward. Faster.

Questions

Moshe Ferber

@: moshe (at) onlinecloudsec.com
w: www.onlinecloudsec.com
in: www.linkedin.com/in/MosheFerber
t: @FerberMoshe



Nir Valtman

@: nir.Valtman (at) ncr.com
w: www.ncr.com | www.valtman.org
in: www.linkedin.com/in/valtmanir
t: @ValtmaNir