Automating Cloud Forensics Lab Provisioning

Tim Ip
Security Coordination Center Incident Response
About Me - Tim Ip

Adobe Incident Response
- Work at Ground Zero
- Focus on Incident Response Automation to make Incident Responders’ lives easier

Previously DevSecOps Engineer in the Life Sciences industry, Security Architect at a University, Security Big Data Consultant at Big 4
- Focus on Big Data and Automation
- Purple Teaming (Offensive Security, Detection Engineering and Big Fan of Splunk/Sysmon)
- Splunk-er for nearly 10 years

Director of Monitoring, Global Collegiate Penetration Testing Competition (https://cp.tc)
- Managing monitoring infrastructure
- Detection Engineering, Threat Hunting in competitive environment
What does a Forensics Lab look like?
What does a Forensics Lab look like?
What does a Forensics Lab look like?
Are you able to successfully perform forensics for Cloud Compute Workloads?
What will this talk cover?

Our approach in handling forensics for Cloud Compute Workloads (Virtual Machines)

- AWS: Elastic Compute Cloud (EC2)
- Azure: Virtual Machines
- Google Cloud: Compute Engine
Forensics Lab
Problems with Shared Forensics Environments
Problems with Data Acquisitions

- Native Cloud Logs (CloudTrail, IAM, VPC Flow) from SIEM
- Disk Images
- Memory Dumps
Workloads Everywhere

Where should I go?
Workloads Everywhere
Workloads Everywhere

SO YOU MIGRATED TO THE CLOUD

AND THOUGHT IT WOULD BE CHEAP?

Monthly Budget

Forgotten instances
Requirements

- **Automate to create a forensics environment SUPER fast**
  - Using available forensics tools
  - Secure and hardened

- **Available in most regions across major cloud service providers (AWS, Azure and GCP)**

- **Make our lab environment ephemeral to save money**
  - Only up during incident
  - Create when incident starts, tear down when incident ends

- **Provide a way to easy archive forensics artifacts to permanent storage**

- **Allow collaboration for Forensics Lab development**

**Solution:**
Automate Forensics Lab setup using Infrastructure as Code (IaC)
Terraform

Provision Forensics VM Cloud Resources (VPC, EC2 Instance, Security Group, IAM Role, etc.)

module "ec2_instance" {
  source = "terraform-aws-modules/ec2-instance/aws"
  name = "single-instance"

  instance_type  = "t2.micro"
  key_name      = "user1"
  monitoring    = true
  vpc_security_group_ids = ["sg-12345678"]
  subnet_id     = "subnet-eddcdzz4"

  tags = {
    Terraform = "true"
    Environment = "dev"
  }
}
Configure Forensics VM

- name: Play Web - Create apache directories and username
  in web servers
  hosts: webservers
  become: yes
  become_user: root
  tasks:
  - name: create username apacheadm
    user:
      name: apacheadm
      group: users, admin
      shell: /bin/bash
      home: /home/weblogic
Requirements

- No need to learn Terraform/Ansible before using it
- Simple and fast (Single Step) to spin up/tear down Forensics Lab environment

Solution:
A Wrapper for Terraform/Ansible
Forensics VM (FVM)
Forensics VM Wrapper

Orchestrate Terraform and Ansible based on requirements provided by Incident Responders

- Cloud Access Key
- SSH Key
- Wrapper Script
- Forensics VM Configuration

```
[sample_aws]
incident_name = delawareaws
cloud_provider = aws
environment = generic
region = us-east-1
az = us-east-1b
sshkey_public_path = ~/.ssh/id_rsa.pub
disk_size_gb = 1000
instance_type = t3.2xlarge
plugins =
  all_gatherfacts, all_createmountpoint, all_docker, all_forensics, aws_forensics, all_tsk, all_volatility, all_tmout, all_maxlogins, all_addsshkeys, all_sethostname, all_falcon, all_splunk, aws_s3upload
```
## Forensics VM Configuration (forensicsvm.conf)

- **A configuration file to manage multiple Forensics VMs**
- **Stanza:** A section of a configuration file. Stanzas begin with a text string enclosed in brackets and contain one or more configuration parameters defined by key/value pairs.
- **Define incident name, VM location (Cloud provider/environment/region/az), disk space, plugins, etc.**

### Example Stanzas

<table>
<thead>
<tr>
<th>Incident</th>
<th>Cloud Provider</th>
<th>Environment</th>
<th>Region</th>
<th>AZ</th>
<th>SSH Key Public Path</th>
<th>Disk Size GB</th>
<th>Instance Type</th>
<th>Plugins</th>
</tr>
</thead>
<tbody>
<tr>
<td>Incident 1</td>
<td>AWS</td>
<td>Adobe</td>
<td>US-West-2</td>
<td>US-West-2a</td>
<td>~/.ssh/id_rsa.pub</td>
<td>500</td>
<td>t3.2xlarge</td>
<td>all_gatherfacts, all_createmountpoint, all_docker, all_forensics, aws_forensics, all_tsk, all_volatility, all_tmout, all_maxlogins, all_addsshkeys, all_sethostname, all_falcon, all_splunk, aws_s3upload</td>
</tr>
<tr>
<td>Incident 2</td>
<td>Azure</td>
<td>Adobe</td>
<td>WestUS2</td>
<td></td>
<td>~/.ssh/id_rsa.pub</td>
<td>500</td>
<td>Standard_B4ms</td>
<td>all_gatherfacts, all_createmountpoint, azure_mountdisk, all_docker, all_forensics, all_tsk, all_volatility, azure_prepprofileenv, all_tmout, all_maxlogins, all_addsshkeys, all_sethostname, all_falcon, all_splunk, azure_allowsplunkwebfw</td>
</tr>
<tr>
<td>Incident 3</td>
<td>GCP</td>
<td></td>
<td>US-West1</td>
<td>US-West1-a</td>
<td></td>
<td>500</td>
<td>e2-standard-8</td>
<td>all_gatherfacts, all_docker, all_forensics, all_tsk, all_volatility, all_tmout, all_maxlogins, all_splunk</td>
</tr>
</tbody>
</table>

```
[incident_1]
incident_name = this_is_a_template
cloud_provider = aws
environment = adobe
region = us-west-2
az = us-west-2a
sshkey_public_path = ~/.ssh/id_rsa.pub
disk_size_gb = 500
instance_type = t3.2xlarge
plugins = all_gatherfacts,all_createmountpoint,all_docker,all_forensics,aws_forensics,all_tsk,all_volatility,all_tmout,all_maxlogins,all_addsshkeys,all_sethostname,all_falcon,all_splunk,aws_s3upload

[incident_2]
incident_name = this_is_a_template
cloud_provider = azure
environment = adobe
region = westus2
ssh_login_id = sccforensics
sshkey_public_path = ~/.ssh/id_rsa.pub
disk_size_gb = 500
instance_type = Standard_B4ms
plugins = all_gatherfacts,all_createmountpoint,azure_mountdisk,all_docker,all_forensics,all_tsk,all_volatility,azure_prepprofileenv,all_tmout,all_maxlogins,all_addsshkeys,all_sethostname,all_falcon,all_splunk,azure_allowsplunkwebfw

[incident_3]
incident_name = this_is_a_template
cloud_provider = gcp
region = us-west1
zone = us-west1-a
disk_size_gb = 500
instance_type = e2-standard-8
plugins = all_gatherfacts,all_docker,all_forensics,all_tsk,all_volatility,all_tmout,all_maxlogins,all_splunk
ssh_login_id = CHANGEME_adobe_com
```
Plugins

- Forensics VM plugins are Ansible tags. By including plugins in configuration files, you can customize your FVM as well as shorten FVM spin-up time.

- Anyone can develop plugins to add more functionality to FVM.

<table>
<thead>
<tr>
<th>Plugin Name</th>
<th>Usages</th>
</tr>
</thead>
<tbody>
<tr>
<td>all_addsshkeys</td>
<td>Add SSH Keys</td>
</tr>
<tr>
<td>all_createmountpoint</td>
<td>Create mount point</td>
</tr>
<tr>
<td>all_docker</td>
<td>Install and configure Docker</td>
</tr>
<tr>
<td>all_falcon</td>
<td>Install EDR</td>
</tr>
<tr>
<td>all_forensics</td>
<td>Create forensics directory structure and install and configure various tools and libraries</td>
</tr>
<tr>
<td>all_gatherfacts</td>
<td>Default - Gather information for Ansible</td>
</tr>
<tr>
<td>all_maxlogins</td>
<td>Adjust maxlogins setting to allow multiple sessions for a single account</td>
</tr>
<tr>
<td>all_sethostname</td>
<td>Configure hostname</td>
</tr>
<tr>
<td>all_splunk</td>
<td>Install Splunk</td>
</tr>
<tr>
<td>all_tmout</td>
<td>Unlock TMOUT restriction</td>
</tr>
<tr>
<td>all_tsk</td>
<td>Install The Sleuth Kit</td>
</tr>
<tr>
<td>all_volatility</td>
<td>Install Volatility</td>
</tr>
<tr>
<td>aws_forensics</td>
<td>Create and install various tools and libraries specific to AWS</td>
</tr>
<tr>
<td>aws_s3upload</td>
<td>Configure AWS Role for S3 Upload</td>
</tr>
<tr>
<td>azure_allowsplunkwebfw</td>
<td>Configure Azure Firewall to allow SplunkWeb traffic</td>
</tr>
<tr>
<td>azure_mountdisk</td>
<td>Mount Forensics volume</td>
</tr>
<tr>
<td>azure_prepprofileenv</td>
<td>Configure Volatility profile compile environment for Azure VM</td>
</tr>
</tbody>
</table>
Single command to spin up/teardown FVM

./forensicsvm create <stanza>

./forensicsvm destroy <stanza>

forensicsvm.conf

[incident_1]
incident_name = incident_1
cloud_provider = aws
environment = adobe
region = us-west-2
az = us-west-2a
sshkey_public_path = ~/.ssh/id_rsa.pub
disk_size_gb = 500
instance_type = t3.2xlarge
plugins =
all_gatherfacts,all_createmountpoint,all_docker,all_forensics,
asw_forensics,all_tsk,all_viability,all_tmout,all_maxlogins,all_addsshkeys,all_sethostname,all_falcon,all_splunk,aws_s3upload
Forensics Pipeline and FVM Lifecycle
Scripts to automatic Triage and Data Archival

Automation Script for fast triage
- Volatility Triage
- Disk Image Triage / Plaso
- Application Triage

Automation Script for data archival
- S3 Archival
- Azure Storage Archival
Demo
Benefits

- Able to create Forensics Labs anywhere, anytime
- Available for popular cloud service providers
  - Analyze data locally: Avoid potential compliance issues (No need to transfer data out from jurisdictions)
- Ephemeral Lab Environment
  - Save $$ (FVM only up during incident)
  - Fresh environment at start
- Encourage contributions and knowledge sharing
- Standardize workflow and formalize forensics pipeline
Future Development

- Expand coverage to spin-up Windows FVM
- Add more forensics triage tools to improve our triage efficiency
- Automated FVM provisioning through ticketing system
Takeaways

- Use Infrastructure as Code (IaC) to provision lab environment
- Create forensics pipeline with well-defined workflow/process
- Test the pipeline regularly
- Create a platform to encourage contributions and knowledge sharing