Manipulating CI/CD Pipeline (version 1.0)

**Cloud Service Label: IaaS, PaaS**

Description

Many cloud providers support automated integration of various Continuous Integration/Continuous Deployment tools such as Jenkins, GitLab, Travis CI, or Circle CI. However, there are downsides for faster and more efficient code releases through a CI/CD pipeline. Some include oversight and lack of testing for deployment systems. An attacker who gains permission to these services could add code that as long as it passes quality checks will likely be incorporated in cloud applications allowing code execution with the permissions of the application. Adversaries may even attempt to sign code in an attempt to thwart CI/CD security checks.

Examples

|  |  |
| --- | --- |
| **Name** | **Description** |
| Jenkins CVE’s | There are many related vulnerabilities related to Jenkins modules and plug-ins. They include [CVE-2016-0788](https://cve.mitre.org/cgi-bin/cvename.cgi?name=2016-0788), [CVE-2017-2652](https://cve.mitre.org/cgi-bin/cvename.cgi?name=2017-2652), [CVE-2015-7539](https://cve.mitre.org/cgi-bin/cvename.cgi?name=2015-7539), and many others. Full list can be found on the [CVE website](https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=jenkins). |
| Configuration Files | At DEFCON 25, spaceb0x covered environment variable manipulation, command injection, and malicious resource provisioning by editing public configuration files and sending them to build or deployment servers. |
| CIDER | Spaceb0x presented the Continuous Integration and Deployment Exploiter (or CIDER) at DEFCON 25, which is a framework aimed to exploit CI/CD pipelines through GitHub pull requests of both public and private source code repositories. |
| Rotten Apple | Rotten Apple is a CI/CD audit framework used to identify if the root user is being used to build projects and if attackers can deploy malicious code to steal API keys, to pivot to private networks, to authenticate using GitHub credentials, to create reverse shells, to exfiltrate data, to access other projects on the same server or to steal SSH keys. The framework also has an attack mode, which can be used for penetration testing. |

Mitigations

|  |  |
| --- | --- |
| **Mitigation** | **Description** |
| Limit Permissions for Applications | If code is inserted into a CI/CD pipeline a cloud administrator is unlikely to know. By ensuring applications have only the permissions absolutely required will minimize the impact of compromise. |
| Patches and Updates | Always ensure applications used in the pipeline and the CI/CD pipeline itself are updated and patched to their most recent versions. |
| Zero Trust | Disable transitive/implied trust between applications. |

Detection

There is the potential that a code review of an application or hash comparison of application files from some trusted repo might detect additions to code. Network sockets and connections created by code could be detected by network flow logs. Scanning, monitoring, and alerting tools built into the pipeline can be utilized to notify of any malicious changes or new vulnerabilities.

|  |  |
| --- | --- |
| **Detection of unusual behavior after exploit** | **Description** |
| Create Log Metric Filters and Alarms for AWS | To create a metric filter and alarm:   1. Create a metric filter that checks for IAM policy changes and the *<cloudtrail\_log\_group\_name>* 2. Create an SNS topic 3. Create an SNS subscription to the above topic 4. Create an alarm associated with the filter and SNS topic created in steps 1 and 2 respectively |
| Monitor Activity in AWS Account | Various services in AWS offer logging features that allow for detection capabilities. These include CloudFront, CloudTrail, CloudWatch, Config, and S3. |
| Monitor for Suspicious Activity in Azure | Azure AD can generate anomaly reports than can be run on a daily basis. Azure AD Identity Protection show current risks in its dashboard and provides daily email summary notifications. Policies can also be configured to alert to specific issues. |
| Create Log Metric Filters and Alarms for CloudTrail | To create a metric filter and alarm:   1. Create a filter that checks for CloudTrail changes and the specific *<cloudtrail\_log\_group\_name>* 2. Create an SNS topic that the alarm will notify 3. Create an SNS subscription to the above topic 4. Create an alarm associated with the filter from step 1 and SNS topic in step 2 |
| Create Activity Log Alerts in Azure | To create log activity alerts for deletion in the Azure Console:   1. Navigate to *Monitor’ / ‘Alerts* 2. Select *Manage alert rules* 3. Click on the Alert *Name* where Condition contains *operationName equals Microsoft.Network/networkSecurityGroups/securityRules/delete* 4. Hover a mouse over *Condition* to ensure it is set to *Whenever the Administrative Activity Log “Delete Security Rule (networkSecurityGroups/securityRules)” has “any” level with “any” status and event is initiated by “any*” |
| Create, View, and Manage Activity Alerts in Azure Monitor | To create a log alert in the Azure portal:   1. Select **Monitor -> Alerts** 2. Select **New alert rule** of the **Alerts** window 3. Provide information in **Define alert condition** 4. Provide details in **Define alert details** 5. Specify action group for new alert rule under **Action group**, or create a new action group with + **New group** 6. Select **Yes** for the **Enable rule upon creation** option 7. Select **Create alert rule**   To view and manage alerts:   1. Select **Monitor -> Alerts -> Manage alert rules** 2. Select the rule you want to modify and double-click to edit the rule options 3. Click **Save** |
| Azure Resource Manager Templates | Azure Resource Manager templates in the format of JSON files that can be used to configure metric alerts in Azure Monitor. These templates can be used for simple static and dynamic threshold metric alerts, availability tests, and monitoring multiple resources. |
| Enable CloudTrail across all regions in AWS | To enable CloudTrail across all regions:   1. Sign into the AWS Management Console and open the CloudTrail console 2. Click on *Trails* 3. Set necessary Trails to All option in the I column 4. Click on a trail via the link *Name* column 5. Set *Logging* to *ON* 6. Set *Apply trail to all regions* to *Yes* |
| Configure log profile to capture activity logs for all regions in Azure | To set up activity logs for all regions:   1. Navigate to Azure console 2. Go to *Activity log* 3. Select *Export* 4. Select *Subscription* 5. Check *Select all* in *Regions* 6. Select *Save* |

References

1. <https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=jenkins>. Accessed July 13, 2020.
2. https://thenewstack.io/poorly-configured-ci-cd-systems-can-be-a-backdoor-into-your-infrastructure/. Accessed July 06, 2020.
3. https://www.youtube.com/watch?v=mpUDqo7tIk8. Accessed July 06, 2020
4. https://github.com/spaceB0x/cider. Accessed July 07, 2020
5. https://github.com/claudijd/rotten\_apple. Accessed July 07, 2020