Stealing Valid Credentials from Local Machines (version 1.0)

**Cloud Service Label: IaaS**

Description

Adversaries may steal the credentials of cloud administrators using well-known endpoint credential harvesting techniques. Compromised credentials may be used to bypass access controls placed on various resources on systems within the network and may even be used for persistent access to remote systems and externally available cloud services, such as cloud storage services. Compromised credentials may also grant an adversary increased privilege to specific cloud systems or resources.

It is common for adversaries to utilize stolen private keys, to legitimately connect to remote IaaS environments via  [Remote Services](https://attack.mitre.org/techniques/T1021). AWS routinely assigns remote keys to every new VM created. These keys are usually downloaded and stored on the local box of the user who is performing the VM creation task. A local exploit of this user’s workstation now also compromises access to whatever VM’s this user has created.

Local web browsers of cloud admin console users cache access tokens that enable quick reconnection if a browser window is accidentally closed. However these tokens may also be stolen from local machines to grant cloud console access to adversaries who have a presence on the local workstation of that user.

The overlap of local workstation access, credentials, and permissions across a network of local workstations and Cloud systems is of concern because the adversary may be able to pivot across accounts and systems to reach a high level of access (i.e., user plane to control plane) to bypass access controls set within the enterprise.

Examples

|  |  |
| --- | --- |
| **Name** | **Description** |
|  |  |

Mitigations

|  |  |  |
| --- | --- | --- |
| **Mitigation** | | **Description** |
| Audit | | Routinely audit source code, application configuration files, open repositories, and public cloud storage for insecure use and storage of credentials. |
|  | AWS | To perform an audit via AWS it is suggested to review information such as account details (credentials, users, groups, roles, etc), mobile applications, EC2 configurations, policies, and account activity. How to audit these different factors can be found in detail at: **https://docs.aws.amazon.com/general/latest/gr/aws-security-audit-guide.html.** |
|  | Azure | To perform an audit via Azure an administrator can review the audit logs that are recorded under Azure’s monitoring for active directory. The audit logs allow for filtering, as well as looking at users, groups, and enterprise specific information. Full details on how to access this information can be found at: **https://docs.microsoft.com/en-us/azure/active-directory/reports-monitoring/concept-audit-logs.** |
|  | GCP | To perform an audit via GCP the logs can be reviewed. GCP breaks this down into three categories; admin activity, data access, and system events. The audit logs can be viewed a few different ways- the console, API, or gcloud. Full details on how to view these logs, how to export, and for how to configure the retention period can be found here: **https://cloud.google.com/logging/docs/audit.** |
| Consider Using Cloud Shells | | Azure offers admins a command line interface directly from the web console that already has been authenticated. This potentially eliminates the need for admins to download cloud credentials locally to their workstation. |
| Filter Network Traffic | | Cloud service providers support IP-based restrictions when accessing cloud resources. Consider using IP whitelisting on cloud-based systems along with user account management to ensure that data access is restricted not only to valid users but only from expected IP ranges to mitigate the use of stolen credentials to access data. |
|  | AWS | An AWS environment can be configured with network ACLs (access control lists) to allow or deny inbound and outbound traffic. This can be accomplished by accessing Amazon VPC and navigating to either inbound or outbound rules depending on the rule the user wishes to add and they can be added, removed, or edited from that panel. Full details about ACLs and how to add rules in AWS can be found here: **https://docs.aws.amazon.com/vpc/latest/userguide/vpc-network-acls.html.** |
|  | Azure | In Azure storage resources can be tied exclusively to a particular virtual network reducing the chances that it can be accessed externally or from other cloud assets. This can be done multiple ways including the Azure Portal, Azure PowerShell, and Azure CLI (Command Line Interface). Depending on the method used to implement this the procedure can vary, but will include the need to create a security group, create a network security group, associate that network security group with a specific subnet and then create security rules that are associated to the inbound and outbound rules for that subnet. Full details on how to configure this utilizing the various methods can be found below:  Azure Portal: **https://docs.microsoft.com/en-us/azure/virtual-network/tutorial-filter-network-traffic**  Azure PowerShell: **https://docs.microsoft.com/en-us/azure/virtual-network/tutorial-filter-network-traffic-powershell**  Azure CLI: **https://docs.microsoft.com/en-us/azure/virtual-network/tutorial-filter-network-traffic-cli** |
| Honey Tokens | | Create accounts credentials with no inherent rights that will be noticed in CloudTrail or Monitor and indicate malicious activity. |
| Key Policies | | SSH keys should be updated periodically and properly secured. In cloud environments, consider rotating access keys within a certain number of days for reducing the effectiveness of stolen credentials. |
| [Multi-factor Authentication](https://attack.mitre.org/mitigations/M1032) | | Use multi-factor authentication for user and privileged accounts. Do not manage Cloud portals from machines that perform user email and web browsing tasks. All users should be required to utilize two factor authentication. |
|  | AWS | This can be enforced by first creating a policy that would prohibit actions except those that allow a user to change their password or manage 2FA, then attaching a policy to a group that includes all user accounts where they can be allowed all access if they sign in with 2FA. Once these actions are completed it should be tested to verify the access is given correctly. To see full details on how to complete this view AWS documentation at: **https://docs.aws.amazon.com/IAM/latest/UserGuide/tutorial\_users-self-manage-mfa-and-creds.html.** |
|  | Azure | This can be done by creating a MFA registration policy. It can than be assigned to all users (with the ability to exclude some if need be, but is not recommended). Make sure once the policy is created and added to users that it is then being enforced, once enforced it should be tested for verification. To see full details on how to complete this view Azure documentation at: **https://docs.microsoft.com/en-us/azure/active-directory/identity-protection/howto-identity-protection-configure-mfa-policy.** |
|  | GCP | This can be done by first enabling it on the current account being used by admin to assign the roles, then enable two factor on an instance by instance or project by project basis, then assigning the requirements based on IAM roles and applying it to all users. To see full details on how to complete this view Azure documentation at: **https://cloud.google.com/compute/docs/oslogin/setup-two-factor-authentication.** |
| [Privileged Account Management](https://attack.mitre.org/mitigations/M1026) | | For IaaS Windows server VM’s, audit domain and local accounts as well as their permission levels routinely to look for situations that could allow an adversary to gain wide access by obtaining credentials of a privileged account. Use RBAC to limit the impact anyone account can have on your cloud assets. General “GOD” level accounts that have access to all cloud resources should never be used for daily administration within the cloud. Rather use accounts that have been delegated admin access to certain resources. In Azure for example, using an account with the owner role at the subscription level for every day tasks is a risky proposition. Once the infrastructure has been created and refined these credentials are no longer required for most admin tasks. Limit credential overlap across systems to prevent access if account credentials are obtained. |
|  | AWS | To manage the access that privileged accounts have on the AWS cloud system to only allow administrators to perform administrative tasks on such accounts can be accomplished utilizing limited IAM administrator accounts. To configure this the administrator would have two accounts; one would have administrative rights and no basic access while the other account has basic access with no administrative rights. To limit the administrative account the IAM limited administrator would be used. This is done by creating a policy that gives a user admin rights, but disallows the other actions using the AWS command line interface. This is outlined at: [**https://aws.amazon.com/blogs/security/how-to-create-a-limited-iam-administrator-by-using-managed-policies/**](https://aws.amazon.com/blogs/security/how-to-create-a-limited-iam-administrator-by-using-managed-policies/)**.** |
|  | Azure | To manage the access that privilege accounts have on the Azure cloud system to only allow administrators to perform administrative tasks on such accounts can be accomplished utilizing limited IAM administrator accounts. To configure this the administrator would have two accounts; one would have administrative rights and no basic access while the other account has basic access with no administrative rights. To limit the administrative account the specific administrative needs can be picked from a number of options available (Azure DevOps Administrator, Billing Administrator, Cloud Application Administrator, etc.) These different options can be edited to fit the needs and limit the basic access. This is outlined at: [**https://docs.microsoft.com/en-us/azure/active-directory/users-groups-roles/directory-assign-admin-roles**](https://docs.microsoft.com/en-us/azure/active-directory/users-groups-roles/directory-assign-admin-roles)**.** |
|  | GCP | To manage the access that privilege accounts have on the Azure cloud system to only allow administrators to perform administrative tasks on such accounts can be accomplished utilizing limited IAM administrator accounts. To configure this the administrator would have two accounts; one would have administrative rights and no basic access while the other account has basic access with no administrative rights. To limit the administrative account pre-defined administrator accounts can be used (mobile admin, Google voice admin, help desk admin, etc.). These accounts can be used with their pre-defined settings, or modified depending on specific use cases. These can limit access to basic functionality needed. This is outlined at: **https://support.google.com/a/answer/2405986?hl=en.** |
| [User Account Management](https://attack.mitre.org/mitigations/M1018) | | Ensure users and user groups have appropriate permissions for their roles through Identity and Access Management (IAM) controls. Configure user permissions, groups, and roles for access to cloud-based systems. Implement strict IAM controls to prevent access to systems except for the applications, users, and services that require access. Consider using temporary credentials that are only good for a certain period of time in cloud environments to reduce the effectiveness of compromised accounts. |
| Use a Bastion Host | | Accessing IaaS resources for administration is best done from within the cloud by means of a bastion server that preferably resides in the same VPC/Virtual network as the IaaS assets. By restricting all external network traffic except this bastion host from accessing privileged ports on IaaS assets, the chance of damage done by exploited accounts is diminished. |

Detection

|  |  |  |
| --- | --- | --- |
| **Detection** | | **Description** |
| Monitoring | | Look for suspicious account behavior across systems that share accounts, either user, admin, or service accounts. Examples: one account logged into multiple systems simultaneously; multiple accounts logged into the same machine simultaneously; accounts logged in at odd times or outside of business hours. Activity may be from interactive login sessions or process ownership from accounts being used to execute binaries on a remote system as a particular account. Correlate other security systems with login information (e.g., a user has an active login session but has not entered the building or does not have VPN access). |
|  | AWS | Various services in AWS offer logging features that allow for detection capabilities. These include CloudFront, CloudTrail, CloudWatch, Config, and S3.  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 |
|  | 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.  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** |
| Auditing | | Perform regular audits of cloud accounts to detect accounts that may have been created by an adversary for persistence. Checks on these accounts could also include whether default accounts such as Guest have been activated. These audits should also include checks on any appliances and applications for default credentials or SSH keys, and if any are discovered, they should be updated immediately. |
|  | AWS | To perform an audit via AWS it is suggested to review information such as account details (credentials, users, groups, roles, etc), mobile applications, EC2 configurations, policies, and account activity. How to audit these different factors can be found in detail at: [**https://docs.aws.amazon.com/general/latest/gr/aws-security-audit-guide.html**](https://docs.aws.amazon.com/general/latest/gr/aws-security-audit-guide.html)**.** |
|  | Azure | To perform an audit via Azure an administrator can review the audit logs that are recorded under Azure’s monitoring for active directory. The audit logs allow for filtering, as well as looking at users, groups, and enterprise specific information. Full details on how to access this information can be found at: [**https://docs.microsoft.com/en-us/azure/active-directory/reports-monitoring/concept-audit-logs**](https://docs.microsoft.com/en-us/azure/active-directory/reports-monitoring/concept-audit-logs)**.** |
|  | GCP | To perform an audit via GCP the logs can be reviewed. GCP breaks this down into three categories; admin activity, data access, and system events. The audit logs can be viewed a few different ways- the console, API, or gcloud. Full details on how to view these logs, how to export, and for how to configure the retention period can be found here: [**https://cloud.google.com/logging/docs/audit**](https://cloud.google.com/logging/docs/audit)**.** |

References

1. <https://cloud.google.com/logging/docs/audit>. Accessed July 30, 2020
2. <https://docs.aws.amazon.com/general/latest/gr/aws-security-audit-guide.html>. Accessed July 30, 2020
3. <https://docs.microsoft.com/en-us/azure/active-directory/reports-monitoring/concept-audit-logs>. Accessed July 30, 2020