UTILIZING CLOUDCHECKR FOR SECURITY

A guide to security in your AWS Environment

Abstract

This document outlines steps to properly secure your AWS environment using CloudCheckr. We cover CloudCheckr use cases ranging from monitoring CloudTrail to following security best practices.

CloudCheckr Support

support@cloudcheckr.com

Table of Contents

Using CloudCheckr to Secure Amazon Web Services

The use of Infrastructure as a Service (IaaS) presents many challenges in regards to security. Using IaaS requires rethinking how applications and infrastructure are secured and audited. The perspective on how to do this needs to be rethought and security teams need to build new processes and procedures to accomplish tasks such as Perimeter Assessments, Penetration Tests, Vulnerability Assessments, User-Rights Reviews, and Security Audits.

CloudCheckr was purpose built for meeting the new security requirements of IaaS and is designed to address this new model of elastic, auto scaling, and ephemeral public clouds. In this manual we will introduce the most common use cases for CloudCheckr, specifically how to use CloudCheckr to accomplish necessary security tasks.

Task 1: Reviewing your AWS environment for Security Issues

Security teams should be performing initial and periodic reviews of the security configuration and vulnerabilities of each AWS account in their organization. An effective review requires the following:

a) Thorough knowledge of standard security best practices.

b) Understanding the nuances of implementing best practices in the new cloud environment.

c) Organizational policies on what is acceptable for the application/infrastructure.
E.g. what is the organization's backup retention policy? What is the password policy?

d) Risk assessment of the AWS account or the application it is hosting to understand the acceptable security settings.
E.g. some application may require MFA for all access to AWS resources. Other AWS accounts may allow public access to some resources.

A security review should be performed before an application is brought into production. After that, security reviews should be performed periodically ranging from daily to annually.

Using CloudCheckr to review your AWS environment

CloudCheckr provides over 100 security checks for AWS. Out of the box, CloudCheckr will perform a review of the security settings of your AWS management plane and save it into Best Practices results. Access to those results can be reviewed historically to determine when a security issue arose. Users can also manually kick off scans after remediation to verify changes.

Using CloudCheckr, the security team can review security for the entire AWS environment. CloudCheckr will automatically generate and distribute daily reports showing how the environment compares to a prepackaged library of security best practice checks.

However, for exceptionally large or dynamic environments, managing security reviews for all AWS accounts on a daily basis may be overwhelming. In this case, we recommend setting up your

complete AWS environment and monitoring specifically for best practice checks that are marked with an Importance level of High. You can configure CloudCheckr to automatically notify the security team of only those security issues.

CloudCheckr can also be setup to review the security in more depth for specific AWS accounts. This may be easier for a security team to manage than attempting for the entire environment.

Configuring AWS Accounts in CloudCheckr

Within CloudCheckr you can setup your AWS accounts to enable you to view security details, reports, and a complete inventory of resources. You will need to register each of your AWS accounts in CloudCheckr. Start by gathering up a list of all the AWS accounts that you will need to monitor for security issues.

For each of the AWS accounts you will be monitoring, you will need to configure read-only access for CloudCheckr. There are two methods to do this:

1. Create an Access Key and a Secret Key for each AWS account. For instructions on this, click this link: http://support.cloudcheckr.com/getting-started-with-cloudcheckr/adding-credentials-in-cloudcheckr/creating-an-aws-user-group-and-policy/

2. Create a trust relationship from each account to our Cross-Account Role. For instructions on this, click this link: http://support.cloudcheckr.com/getting-started-with-cloudcheckr/adding-credentials-in-cloudcheckr/cross-account-roles/

After configuring an AWS account in CloudCheckr, you can click the Update button and CloudCheckr will begin monitoring your AWS account and building reports based on its findings.

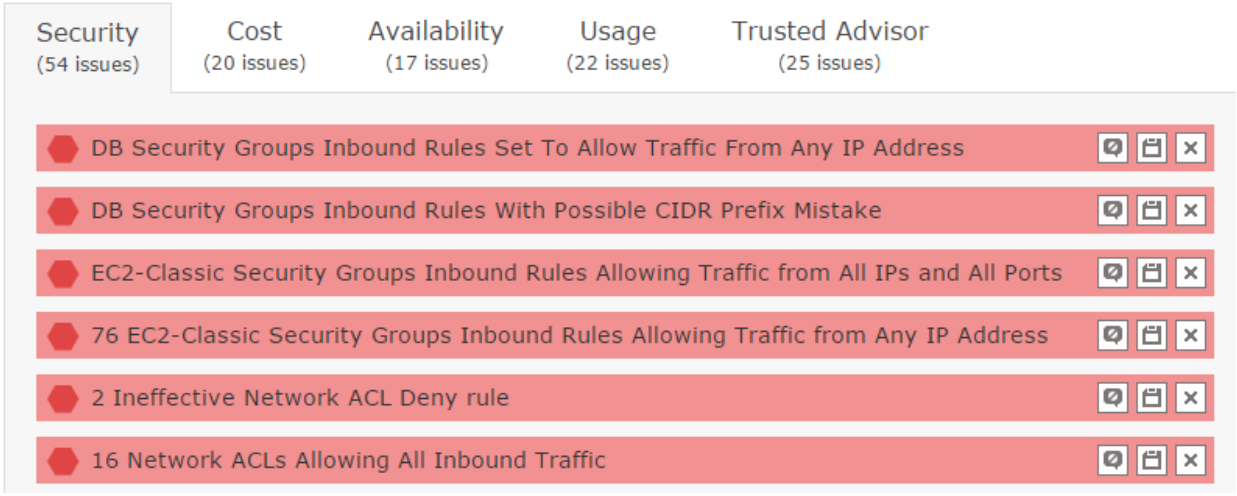
Cross-Account Visibility

CloudCheckr allows you to tag AWS accounts and map those together to create groups of AWS accounts. These groups are known in CloudCheckr as Multi-Account Views. You can also create a Multi-Account View for all AWS accounts, from which you can see all your AWS accounts (and best practice checks) in a single view.

Follow the steps here (http://support.cloudcheckr.com/cloudcheckr-project-tag-userguide/) to get your Multi-Account Views up and running. Once that is completed, best practice checks will be pulled from all the tagged AWS accounts into a single Best Practices report.

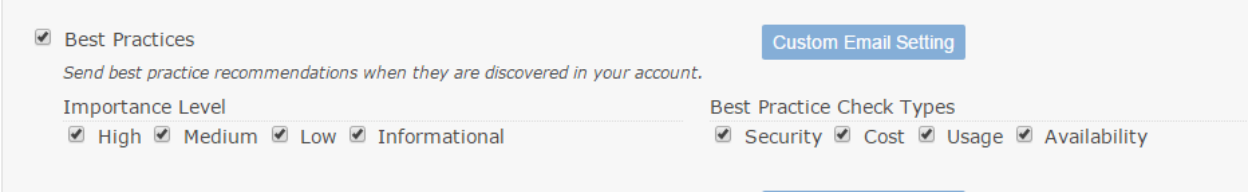
Reviewing Best Practice Checks

The Best Practices reports shows the details of each issue discover. To find the report, navigate to the Best Practice on the left menu and select the Security tab. Best Practice checks are order and color-coded to their importance level.



Configuring Emails for Best Practice Checks

CloudCheckr will send you nightly updates of new violations discovered thru Best Practice checks. You can customize this email by setting the specific email address, time of the email, Importance levels, and check type. To do this, navigate to the Settings option on the left-hand menu and select Email. Select the Daily tab and scroll down until you see Best Practices.



Task 2: Verifying CloudTrail Configuration

CloudTrail provides activity monitoring capability for the AWS management plane. CloudTrail records every call into the AWS API. Everything done in AWS is accomplished using the API, including when tools such as the AWS Management Console are used. So you can be confident any activity taken in AWS is recorded into the CloudTrail logs.

CloudTrail logs are written into an S3 bucket as JSON files. A separate file is written every five minutes. Additionally, a different file is created for each AWS account and each region. Realistically, looking directly into the CloudTrail files is not a practical task. You need to use a tool to consume and understand what is contained in these files. The CloudTrail UI provides basic functionality to look up events for up to seven days, but undoubtedly you will require access to events from prior periods or more powerful search capabilities.

The first task for security professionals is ensuring CloudTrail is enabled properly. This means ensure it is enabled in all AWS accounts and all regions within those accounts. Note: there is a small marginal cost of using CloudTrail, namely the cost of the storage. You can setup lifecycle rules on the S3

bucket to archive to Glacier or delete CloudTrail logs after a period of time to limit these charges. Rarely do these charges exceed a few dollars per month.

It is common for AWS users to setup CloudTrail but enable it ONLY in the regions they are using. It is highly recommended you enable it in every region, whether it is actively used or not. The purpose of auditing is to capture activity including any that is unusual or suspicious. By failing to enable CloudTrail in unused regions, you leave an opportunity for unmonitored activity to occur. There is near zero cost and minimal effort to enable CloudTrail across all regions in an AWS account. As such, it is highly recommended you do so.

For more information on enabling CloudTrail in all regions click here:
http://support.cloudcheckr.com/getting-started-with-cloudcheckr/enabling-cloudtrail/

Using Best Practice Checks to Verify CloudTrail Configuration

One of the easiest ways to keep track of your CloudTrail configuration is by using the CloudCheckr Best Practice checks. The following checks are all centered around ensuring CloudTrail is writing logs correctly and completely.

CloudTrail Not Enabled: Critical. This check will identify if CloudTrail is not enabled at all in an account.

Regions without CloudTrail Enabled: Critical. Determines if one or more regions in an AWS account has not enabled CloudTrail. Strongly recommend you enable CloudTrail in every region.

CloudTrail Delivery Failing: Critical. If CloudTrail is enabled, but for some reason it cannot write the file into the S3 bucket, you will not have audit logs. This can happen because of permissions on the S3 bucket, because the bucket is deleted or moved, etc... If you find this anywhere, investigate and fix. Delivery failing is no better than not enabling CloudTrail.

CloudTrail Global Service Events: Critical. You should register one and only one CloudTrail to receive what are called Global Service Events. These are events that are not associated with a specific region. For instance, IAM activity is not regional and needs to be written into a one of the regional CloudTrail logs. If you find an AWS account that has no region configured to write Global Service Events you are missing many critical security events. If you find AWS accounts that have multiple regions with Global Service Events enabled, you will see duplicate events.

CloudTrail Disabled: Critical. Identifies when someone disables CloudTrail. Anytime you see this, you should review who disabled it and ensure it was re-enabled.

CloudTrail Notification Failing: Medium. The CloudTrail service can be configured to notify when a new log file was written. This is typically used by monitoring services such as CloudCheckr to get notified as soon as new CloudTrail files are written. Without these notifications, monitoring services need to poll for new files on a constant basis. If a notification is failing, you should review and try to fix the issue so that anyone listening for the notification will process the CloudTrail files on a timely basis. It's even possible that a monitoring service is only waiting for CloudTrail notifications and could be missing all your CloudTrail events.

CloudTrail SNS Topic Missing: Same as the previous best practice check. This check sees if CloudTrail is pointing to a SNS Topic that is missing or deleted.

Task 3: Monitoring CloudTrail

Once you have CloudTrail enabled properly, you need to begin monitoring it. CloudTrail logs are valuable for forensic purposes, but it is even more importance to monitor the logs to know when something unusual or suspicious happens.

In order to monitor CloudTrail, you will need to translate the API events into meaningful terms. For instance, to monitor for security group changes you need to look for the following list of AWS API events:

- CreateSecurityGroup
- DeleteSecurityGroup
- AuthorizeSecurityGroupEgress
- AuthorizeSecurityGroupIngress
- RevokeSecurityGroupEgress
- RevokeSecurityGroupIngress

- CreateCacheSecurityGroup
- DeleteCacheSecurityGroup
- AuthorizeCacheSecurityGroupIngress
- RevokeCacheSecurityGroupIngress

- CreateDBSecurityGroup
- DeleteCacheSecurityGroup
- AuthorizeDBSecurityGroupIngress
- RevokeDBSecurityGroupIngress

- CreateClusterSecurityGroup
- DeleteClusterSecurityGroup
- AuthorizeClusterSecurityGroupIngress
- RevokeClusterSecurityGroupIngress

In order to effectively know what to look for, you will need a complete and comprehensive list of events to monitor for. As you see in the above example, the hardest part of monitoring is understanding what to monitor for. AWS often offers multiple events that, from a user's perspective, have very similar appearing outcomes.

CloudTrail Alerts

CloudCheckr has alerting capabilities specifically for CloudTrail events. These can be found under the Alerts section in the left hand navigation bar. We have included built-in alerts which are designed to help you keep track of the security of your account. To enable these, simply flip the toggle to "On". If you click on the alert you can specify an email address, PagerDuty key, or SNS topic that the alert will be sent to. Below is a sample list of our built-in alerts along with their descriptions. Review the list and

enable the built-in alerts that are critical or high importance levels. Review the other built-in alerts to determine if they are important to your security team.

CloudTrail disabled: Critical. The most important CloudTrail event to monitor for is CloudTrail being disabled. Once CloudTrail is disabled you won't be able to see anything going on. If you receive this Alert, you should immediately investigate what is happening. This Alert monitors for anyone calling the AWS API running the commands StopLogging, UpdateTrail, or DeleteTrail.

Note: UpdateTrail can be used cleverly to disable logging by pointing at an invalid S3 bucket or by disabling Global Service Events. So while you may see some false positives from this alert, you should investigate when UpdateTrail is called and make sure any updates are valid.

Unauthorized access attempt: Critical. Detect a user attempting to access a resource to which they are not authorized.

Root account access key created: Critical. The root account should never be used. If you detect someone creating an access key for a root account, you should take appropriate action.

Root account used: Critical. The root account should never be used. Note: AWS often uses the root account when performing internal actions, such as auto scaling. CloudCheckr will filter out such actions and show only true attempts from an end user to use the root account.

Failed login to AWS Management Console: Medium. It may be useful to get notified when a failed attempt is made to log into the AWS Management Console. If you would like to get notifications of failed login attempts, enable this Built-in Alert. Note: You can also copy this built-in alert and indicate a threshold for the number of events and time frame to trigger this alert.

Any security-related events: High. You can setup alerts of any security-related events to notify the security team. This alert monitors for IAM events and VPC events that make changes to the security of your environment. For instance, IAM policy changes or deleting a Network ACL are two examples. This does not monitor for events that only read data. Only events that modify your security posture will trigger this alert.

Security group modified: High. This alert monitors for changes to security groups. This is a subset of the alert "Any security-related events". This does not monitor for events that read data. Only events that modify your security posture will trigger this alert.

IAM policy modified: High. This alert monitors for changes to IAM policies. This is a subset of the alert "Any security-related events". This does not monitor for events that read data. Only events that modify your security posture will trigger this alert.

Task 4: Using CloudTrail Logs to Investigate Activity

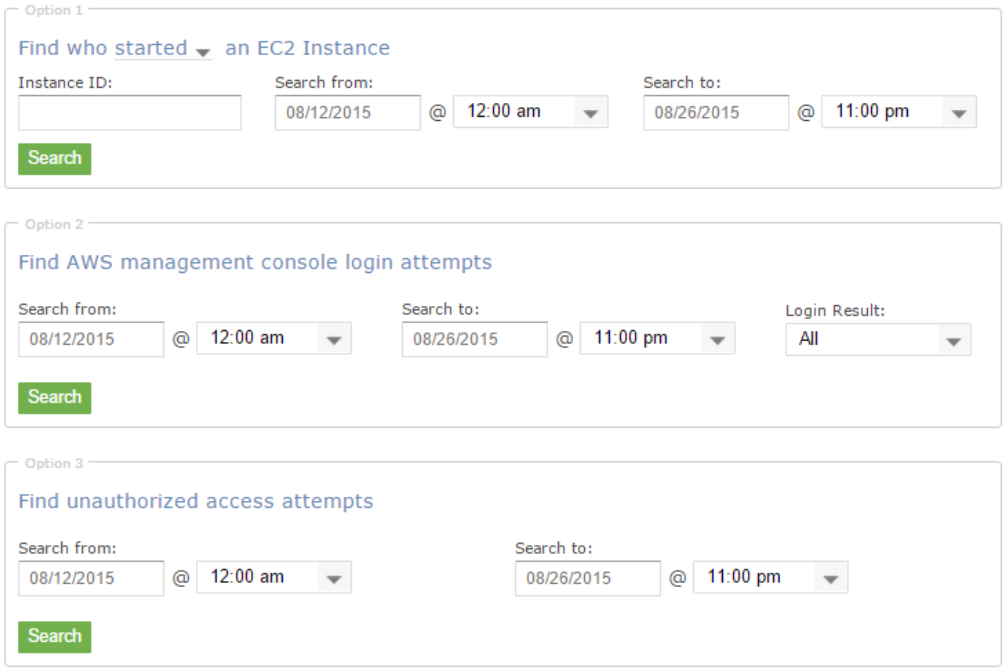
When you find yourself needing to track down who did what in an AWS account, having to use the AWS Management Console or manually read through the CloudTrail logs is impractical. For instance, say you want to see all the IAM users that have been added in the past month to an AWS account. It is entirely impractical to look through all the log files to find the event CreateUser. Search capability like this requires loading data into a database that can be queried.

Another example: assume that you need to find all IAM policy modifications for the past three months in a specific AWS account. To effectively do this you must first gather up all the AWS events that would result in an IAM policy modification. Next, you would have to find a way to search over 200,000 CloudTrail files (a new file every 5 minutes to 9 different regions).

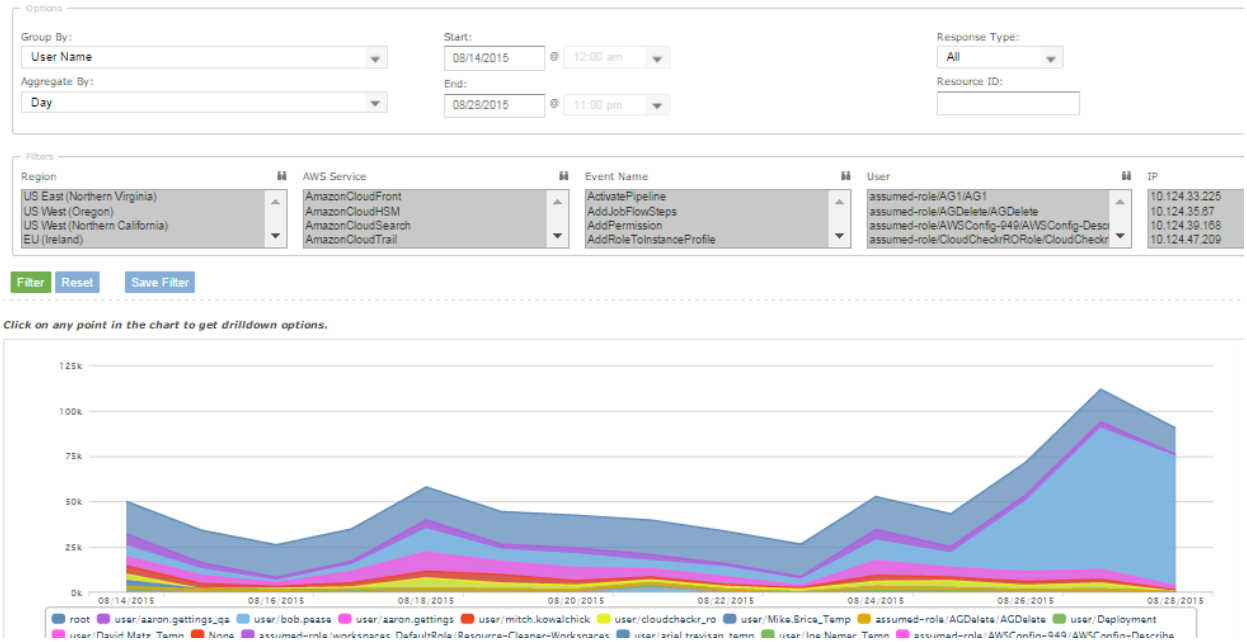
When investigating activity in an AWS account, we recommend starting with the Security/CloudTrail/Common Searches report. Similar to CloudTrail built-in alerts, this screen is compiled of search options which will help guide you to picking the right options to filter by. This page includes the following searches:

- Find who created, started, stopped, terminated an EC2 Instance
- Find AWS management console login attempts
- Find unauthorized access attempts
- Find all activity for a specific IAM user
- Find all activity for a specific IP address
- Find IAM users created in a time period

For example, if we wanted to see the data from the last option, “Find IAM users created during a time period”, select the date and the hour to begin the search and the date and the hour to end the search and select ‘Search’. CloudCheckr will translate this into the event CreateUser. The results will show any IAM user created during that time span.



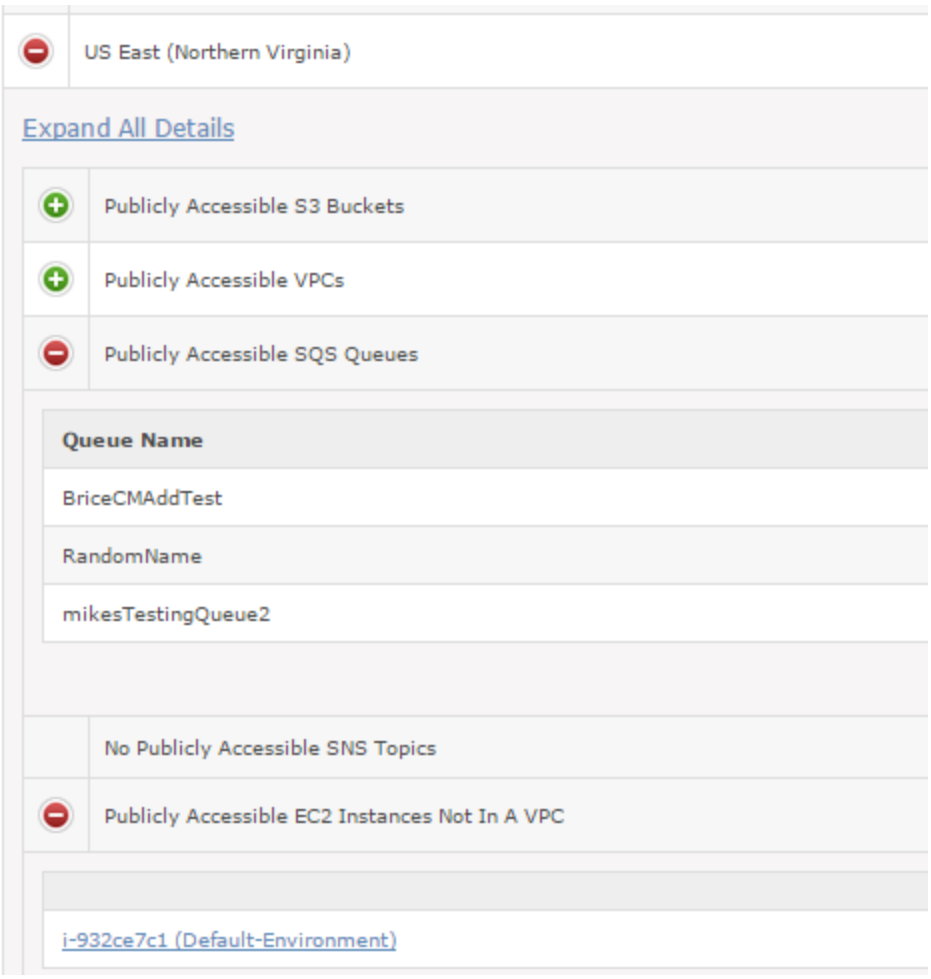
You can also find CloudTrail information by searching directly under Security/CloudTrail/Events. This report gives you the ability to group by different options such as User Name, Event Name, IP Address, and Service. You can choose the time period you would like to search in, the response type as well as a specific resource ID. We also provide filtering options on region, service, event name, IAM users, and IP addresses. This gives you the ability to narrow down results in any way you need.



NOTE: CloudCheckr maintains a complete history of CloudTrail. CloudCheckr records meta-data about the events in order to quantify and search the results, but CloudCheckr does not maintain a complete copy of the CloudTrail event. We recommend you retain the copy of you CloudTrail events in S3 for as long as your security policy requires. You can in addition use Glacier for archiving older CloudTrail log files.

Task 5: Reviewing the Perimeter of your AWS Environment

Moving from a data center to the public cloud requires rethinking how you do assessments of your perimeter security. For this reason CloudCheckr provides the Perimeter Assessment Report. This report will give you information on any publicly accessible resource in each of the available AWS regions. You can find this report under Security/Perimeter Assessment.



Within this report, a + symbol next to a region indicates that there are publicly accessible resources within that region. If you expand a region, it will show you which AWS services have publicly accessible resources and, if you expand the service, it will give you the specific resource and details on the controls around the service and resources.

Resources may be intentionally public. For instance, a web server may well require open access to the Internet. This report helps you to gather the complete list to review and ensure ONLY what is meant to be exposed is. We suggest you review this report and validate any publicly accessible resources are meant to be exposed as such.

Within each region, you will see the list of publicly accessible resources not within a VPC. Verify each and the security groups associated with them to make sure they are appropriately restricted. Ideally you would move resources that can be run from within a VPC into a VPC. These resources types include EC2, RDS, Redshift, Elastic IPs, and ElastiCache. Some resources, such as S3 and DynamoDB, can't be moved into a VPC.

You will also see the VPCs that are publicly accessible. Underneath the VPC you can review the NACL to see which rules are allowing public access on which ports. One level down, CloudCheckr lists the Subnets within each VPC, and the public resources within each subnet. Additionally, CloudCheckr will show the list of security group rules associated with the instances. This is a lot of information, but it

gives you a way to pull all the various controls (VPCs, Security groups, public IP addresses) into a single place to understand if and how access to a resource is restricted (or not).

As we stated earlier, before an application goes into production, you should review the security configuration, starting with Security Best Practice checks. You should also review the privileges and access controls of all the components of the resources. The perimeter assessment report is a good way to start. Get an inventory of the resources from the application team, including the S3 buckets, list of databases, VPCs being used, EC2 instances, auto scaling groups, etc… From this list you use the Perimeter Assessment report to ensure these resources are not publicly accessible unless they are meant to be.

For instance, let's show an example of an application inventory that includes two S3 buckets, one SQS Queue, one RDS database, and five EC2 instances within a VPC. Start by determining the appropriate access of these resources. For instance, one of the S3 buckets might be marketing materials meant to be publicly available to potential customers. The other bucket contains backups of database files. One of the EC2 instances is a webserver and should be available to potential customers over HTTPS on port 443.

Open the Perimeter Assessment, expand the region the S3 buckets are within, and verify that only the one S3 bucket shows up under "Publicly Accessible S3 Buckets". Verify that the SQS Queue does not show up under "Publicly Accessible SQS Queues". Next expand the VPC and ensure the RDS database does not show up under "RDS DB instances with a public IP". Check that only the webserver shows up under "EC2 instances with a public IP" and that the security groups of the instance are limited to port 443.

Exporting Publicly Accessible Resources

This report has three output formats to pull the results out of CloudCheckr.

1) Export – this format gives you a complete export of all the details in the report.
2) Export by Region – this format provides a single line for each region with all the publicly accessible resource within a column in each row.
3) Export by Resource – this format list each resource, the resource type, region, and VPC. This format gives you a list of each publicly accessible resource.

You can use these various reports to detect publicly accessible resources. For instance, you may need a complete list of publicly accessible resources to feed into your vulnerability assessment tools to run scans against. In this case, you can click "Export by Resource" to get a complete list of the public resources to feed into your vulnerability assessment tools.

Using the API to find Publicly Accessible Resources

You can also use the CloudCheckr API to extract the details from the Perimeter Assessment report. You can see a complete description of how to use the API to extract details from the support site:

http://support.cloudcheckr.com/cloudcheckr-api-userguide/cloudcheckr-api-reference-guide/#get_publicly_accessible_resources

The below link is a sample python script that you can use to leverage CloudCheckr's inventory to pull out a list of publicly accessible resources you can then integrate into your vulnerability assessment tools.

http://support.cloudcheckr.com/cloudcheckr-api-userguide/#usecase

Task 6: Auditing Security Groups

Security groups are one of the primary methods used for securing traffic to an EC2 instance, RDS database, Redshift cluster, or ElastiCache cluster. EC2-VPC Security groups can be used to secure any of these resources if they sit in a VPC. If any of these resources are outside of a VPC, you must use security groups that are specific to the resource type. For instance, for RDS you would have to use DB Security Groups.

The two main network security controls in AWS are Security Groups and Network ACLs.

Security groups: Assigned directly to an instance or resource. Rules are stateful, meaning traffic returned from a valid request is allowed irrelevant of the security rules.

Network ACLs: Assigned to an entire subnet in a VPC. Rules are stateless, meaning rules must be defined for return traffic as well.

CloudCheckr provides capabilities to search Security Groups to find ones that are wide open or overly-permissive. An organization may have hundreds of AWS accounts with hundreds of Security Groups. The security team should be reviewing these Security Groups to make sure they are appropriately configured.

The security department can start by reviewing best practice checks. Setup a Multi-Account View to include all AWS accounts, and allow time for the Multi-Account View to collect all results across the accounts. We recommend looking across your entire organization for any issues with the best practice checks below:

- *EC2-VPC Security Groups Inbound Rules Set To All IPs And All Ports*
- *EC2-Classic Security Groups Inbound Rules Allowing Traffic from All IPs and All Ports*
- *DB Security Groups Inbound Rules Set To Allow Traffic From Any IP Address*
- *Redshift Security Groups Inbound Rules Allowing Traffic From Any IP Address*

These checks are finding Security Groups that have no limitations on access at all. A no limitations setting is rarely appropriate. It's highly recommended that you prohibit this as a corporate policy and then monitor for someone inadvertently configuring a group with it. Chances are that many of your AWS accounts will have many of these by default.

The results of these best practice checks look like this:

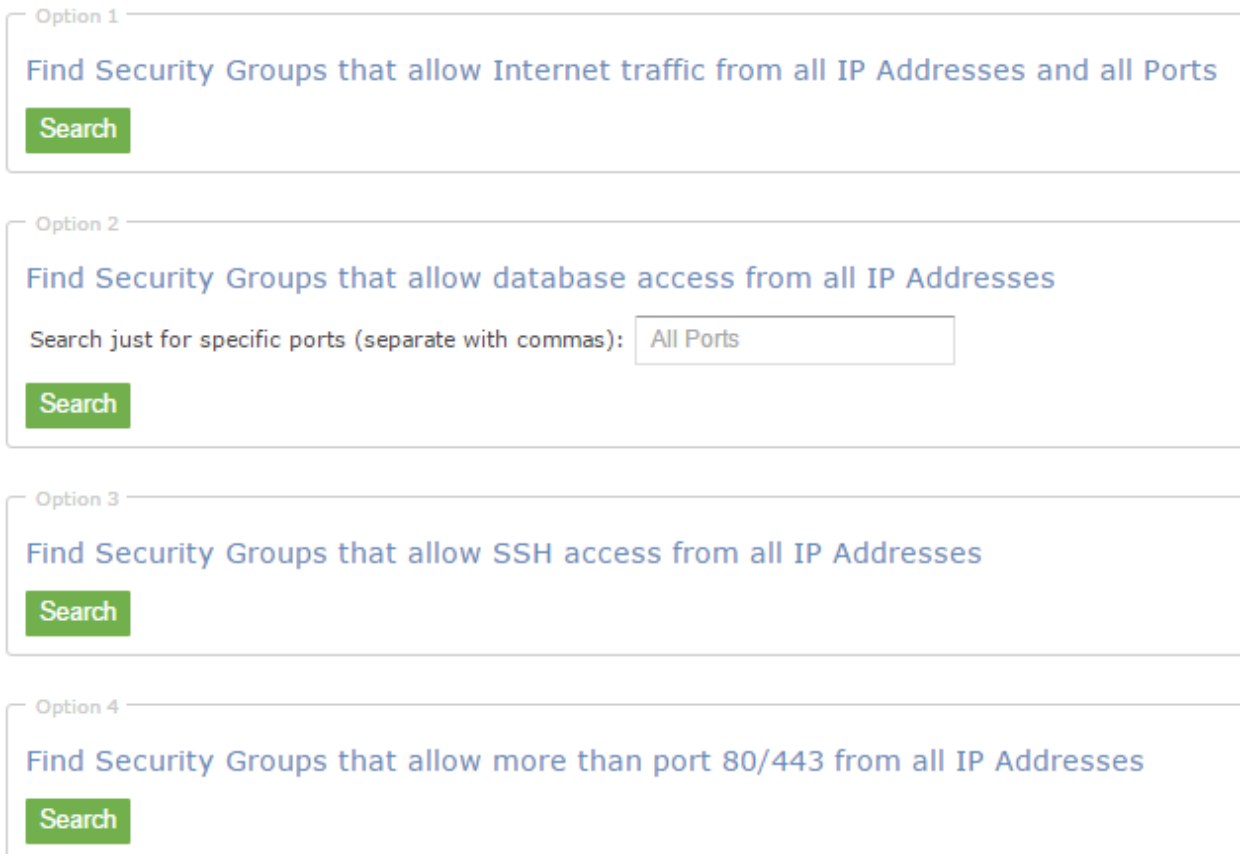
Group: StandaloneSG | ID: sg-4c82c17f | Port Range: 80,443,0-ALL,8-ALL |

Instances using this security group: 2 | Region: US West (Oregon)

The results contain the number of instances using this Security Group so you can prioritize which ones to track down and shutdown first. Many security groups may be overly-permissive, but might not be assigned to any resources. It is recommended you remove these, but you might prioritize these below fixing security groups that are wide-open and have resources assigned to them. As well, for each result you will see "X Ignore Item". If you deem that it is appropriate for a specific security group to be wide open, you can choose to ignore this result. You can always resume monitoring the specific security group later if needed by selecting the "Show Ignore" checkbox above.

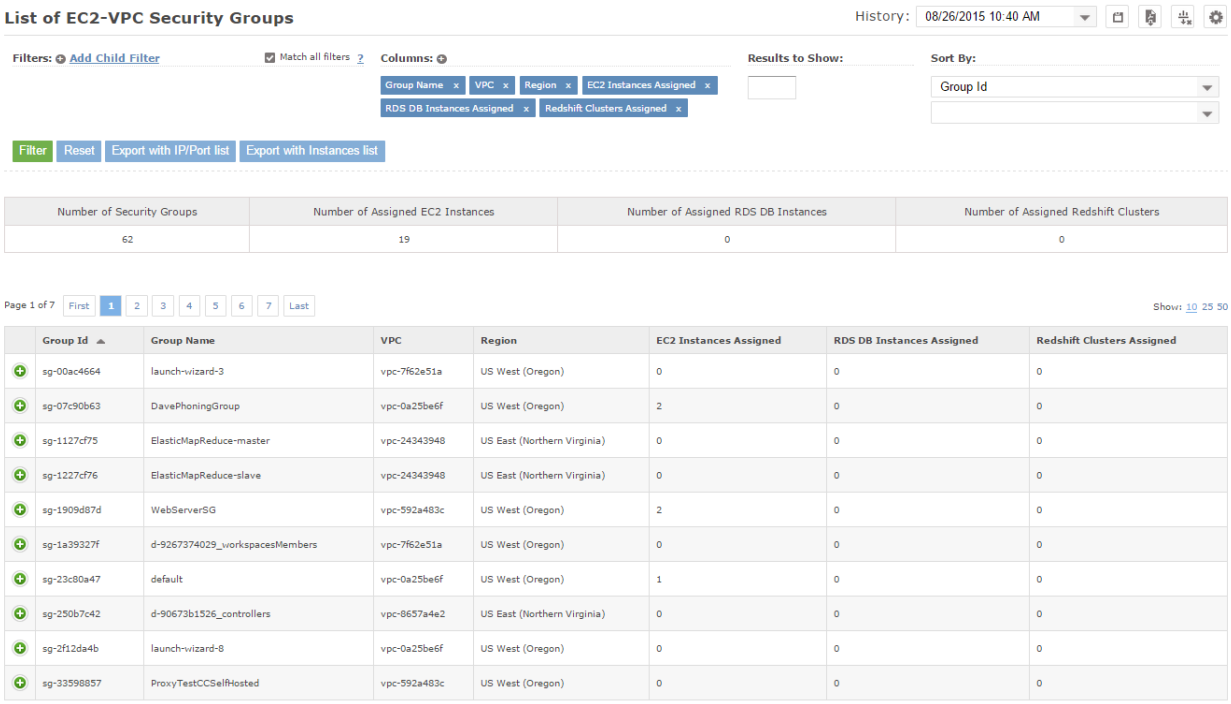
Searching for Specific Security Group Issues

You can also perform ad hoc searches of Security Groups from CloudCheckr. For instance, you should audit your security groups to verify public access to common database ports are shutdown. You can do this within the report Security/Security Groups/Common Searches. Option 2 is labeled "Find Security Groups that allow database access from all IP Addresses". Click "Search" and you will have a list of Security Groups that match the search filter.



You should also review open access from the Internet to port 22. Within the report Security/Security Groups/Common Searches select Option 3 "Find Security Groups that allow SSH access

from all IP Addresses". The complete list of Security Groups that allow incoming traffic over port 22 from ANY IP address will be generated.



List of EC2-VPC Security Groups

History: 08/26/2015 10:40 AM

Filters: Add Child Filter | Match all filters | Columns: Group Name, VPC, Region, EC2 Instances Assigned, RDS DB Instances Assigned, Redshift Clusters Assigned | Results to Show: | Sort By: Group Id

Filter | Reset | Export with IP/Port list | Export with Instances list

Number of Security Groups	Number of Assigned EC2 Instances	Number of Assigned RDS DB Instances	Number of Assigned Redshift Clusters
62	19	0	0

Page 1 of 7 First 1 2 3 4 5 6 7 Last Show: 10 25 50

Group Id	Group Name	VPC	Region	EC2 Instances Assigned	RDS DB Instances Assigned	Redshift Clusters Assigned
sg-00ac4664	launch-wizard-3	vpc-7f62e51a	US West (Oregon)	0	0	0
sg-07c90b63	DavePhoningGroup	vpc-0a25be6f	US West (Oregon)	2	0	0
sg-1127cf75	ElasticMapReduce-master	vpc-24343948	US East (Northern Virginia)	0	0	0
sg-1227cf76	ElasticMapReduce-slave	vpc-24343948	US East (Northern Virginia)	0	0	0
sg-1909d87d	WebServerSG	vpc-592a483c	US West (Oregon)	2	0	0
sg-1a39327f	d-9267374029_workspacesMembers	vpc-7f62e51a	US West (Oregon)	0	0	0
sg-23c80a47	default	vpc-0a25be6f	US West (Oregon)	1	0	0
sg-250b7c42	d-90673b1526_controllers	vpc-8657a4e2	US East (Northern Virginia)	0	0	0
sg-2f12da4b	launch-wizard-8	vpc-0a25be6f	US West (Oregon)	0	0	0
sg-33598857	ProxyTestCCSelfHosted	vpc-592a483c	US West (Oregon)	0	0	0

Task 7: Analyzing Network ACLs

Network ACLs are the firewalls of the VPC. You can set rules that allow or deny access to a port or IP range in a NACL. NACLs have some advantages over Security Groups. For instance, rules applied to NACLs are guaranteed to cover all resources in the subnet, whereas a Security Group applies only to the instances it is explicitly applied to it. Relying on Security Groups exclusively is problematic because someone could inadvertently create an EC2 instance in the VPC and associate an improper Security Group to it, leading to it being compromised. This creates an attack point into your VPC that can be used to leap frog to other instances in the VPC even if they do not have public IP addresses.

The disadvantage of NACLs is that they are stateless. If you allow traffic into a subnet, you must specifically allow the outbound traffic for the ephemeral ports of the return traffic. This can be complex to manage and requires opening large ranges of ports. Read more on this topic here:
http://docs.aws.amazon.com/AmazonVPC/latest/UserGuide/VPC_ACLs.html

CloudCheckr provides capabilities to search NACLs to find ones that are wide open or overly-permissive. An organization may have hundreds of AWS accounts with dozens of VPCs. The security team should be reviewing the NACLs of all VPCs to make sure they are appropriately configured.

The security department can start by reviewing best practice checks. Setup a Multi-Account View to include all AWS accounts and allow time for the Multi-Account View to collect all results across

the accounts. We recommend looking across your entire organization for any issues with the best practice checks below:

- *Network ACLs Allowing All Inbound Traffic*
- *Ineffective Network ACL Deny rule*

The first check finds NACLs that have no limitations on access at all. This is rarely appropriate. It's highly recommended that you prohibit this as a corporate policy and then monitor for someone inadvertently configuring one. Chances are that your AWS accounts will have many of these by default.

The results of this best practice checks look like this:

Network ACL ID: acl-b6b390d3 | VPC: vpc-d5361ab0 | Region: US East (Northern Virginia) | Rule #: 100 | Port Range: ALL | IP Range: 0.0.0.0/0 | Type: ALLOW Inbound

The second best practice check finds NACLs that have security rules which are ineffective or misconfigured. If you discover this, there is a strong likelihood that network traffic that is not intended is being allowed.

Searching for Specific NACL Open Ports

You can also perform ad hoc searches of NACLs from CloudCheckr. For instance, you should audit your VPCs to verify public access to the SSH ports are shutdown. You can do this within the report Security/VPC/Common Searches. Option two is labeled "Find Network ACLs that allow SSH access from all IP Addresses". Click "Search" and you will have a list of NACLs that match the search filter.

Summary

As you can see, moving to the public cloud presents new challenges for a security department. A new set of tasks emerge, and along with those you need new tools to help you perform these tasks. CloudCheckr is purpose built for these use cases, making it simple to keep up with these changes.