Persistent

CyRev

User Guide

Software Version: 1.1.3

Legal Notices

Warranty

The only warranties for products and services are set forth in the express license or service agreements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty of any kind, implied, statutory, or in any communication between them, including without limitation, the implied warranties of merchantability, non-infringement, title, and fitness for a particular purpose. Persistent Systems shall not be liable for technical or editorial errors or omissions contained herein. The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from Persistent Systems or its licensors required for possession, use or copying. No part of this manual may be reproduced in any form or by any means (including electronic storage and retrieval or translation into a foreign language) without prior agreement and written consent from Persistent Systems.

Copyright Notices

Trademark Notices

Persistent Systems are trademarks or trade name or service mark or logo of Persistent. All other brands or products are trademarks, trade name, service mark, logo or registered trademarks of their respective holders/owners thereof.

Disclaimer

The Persistent System products are available and support only the English language.

Table of Contents

Introduction

Persistent Systems' Cyber Recovery Director (CyRev) is an enterprise-class software appliance to discover and recover from ransomware attacks. These attacks require us all to rethink our security posture and we should assume that ransomware will eventually permeate even the most protected environments. CyRev helps you recover from such debilitating ransomware attacks. CyRev uses cyber resilience technology and process from Persistent Systems Ltd. to protect and recover from ransomware attacks in a short span of time with minimal loss of data, minimizing impact to business. CyRev is the most complete and easy-to-use solution available in market.

Ransomware Anatomy

Most ransomware attacks generally unfold in the same fashion and follow the same stages, ultimately ending with the unavailability of a system and/or the system's data.

First, a system becomes infected with malware which allows the attacker access to at least modify some files on the system. As with any malware attack, this initial incursion can occur in a vast number of ways: through vulnerable network services, via a phishing or account password attack, through infected portable storage, etc.

Once the malware has become resident on a system it will attempt to disable any protections or other services on the system that might interfere with the progress of the attack (For Example: antivirus software, backup processes, etc.) It may also attempt to spread itself to other systems via various vectors (network, storage exchange, and credential compromise, for example.)

It will then begin to transform critical data into a form that will be inaccessible by the rightful owner. As with the initial infection this stage can proceed in many different ways. Typically, data is slowly encrypted while still giving access to the unencrypted data to prevent discovery of the attack, either through excessive resource consumption on the system or the discovery of inaccessible data. To this end it also avoids encrypting operating system files that would cause the server or critical services to stop operating. In some cases, the unencrypted data is copied off the system by the attacker for exploitation outside of any ransom demand.

Once enough data has been encrypted (or otherwise made inaccessible) to cause significant inconvenience to the rightful owner, access to the original data is removed usually by deleting it from the system. At this point the attacker will issue a ransom demand to the data owner, insisting on payment before providing a key to unencrypt the data.

Discovery of the attack can occur at any point up to the ransom demand, either through unusual system operation (higher than typical CPU and memory resources, excessive disk usage or disk space consumption, rogue process recognition, etc.) However, ransomware has become proficient at hiding itself from casual discovery in this manner. Additionally, once access to critical data is lost the attack has in effect succeeded: some sort of remediation is required by the system owner in order to avoid eventually paying the ransom demand or to avoid at least some critical service unavailability.

As is the case with other data access interruptions (disk or other hardware failure, loss of a site, etc.) remediation means restoring the data and reconstituting the affected systems and services. As ransomware attacks have evolved, however, attackers have learned to disable or destroy the sources for such remediation. For example, when the system is compromised it's not unusual for credentials for backup or DR systems to be stolen, allowing the malware or attackers to exclude data from being collected for backup or DR; to delete the backup or DR images that would be used for remediation; or to corrupt them to make them unusable or to insert trojan horses to make them instantly re-infectable after recovery. Therefore, it is essential to have a separate, uncompromisable, and incorruptible system for keeping historical images safe from ransomware attacks, beyond just backup or DR.

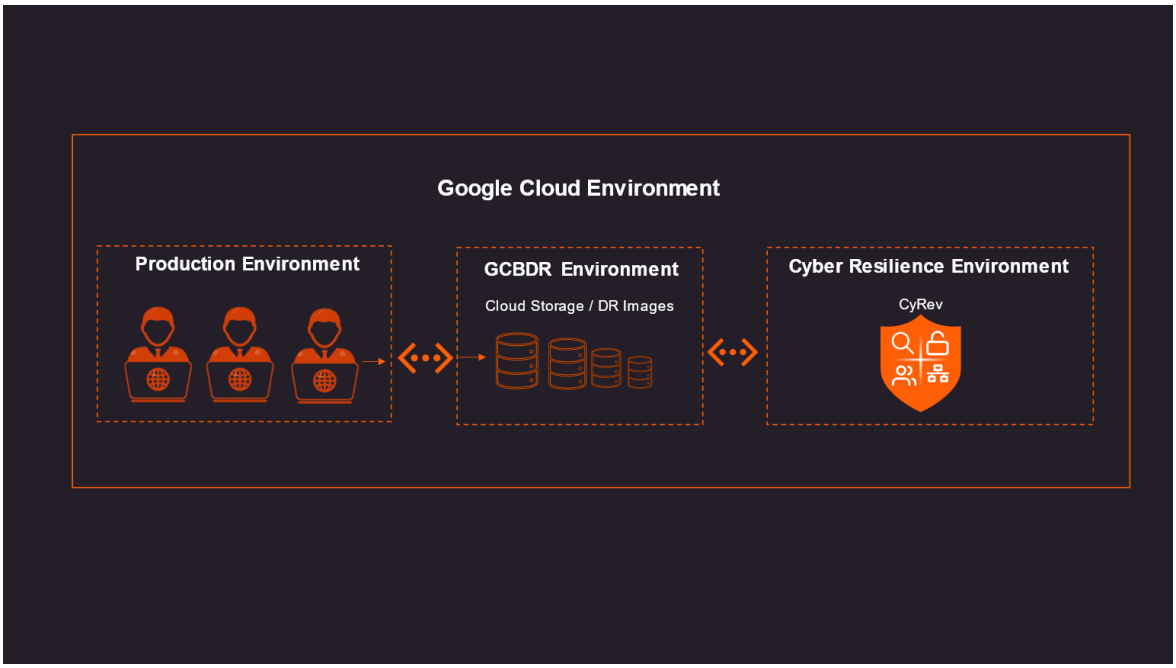
Even with valid historical images for the system and data, reconstruction of infected or inaccessible systems and datasets by hand is itself an arduous task. As images are obtained over time for backup or DR purposes, they will be actually copying infected systems and partially encrypted datasets. Since the process between infection and final ransom demand can take a significant amount of time a pre-infection image is unlikely to have the most desirable (most recent) data. And images taken post-infection will have the malware in them, meaning they must be scanned and scrubbed to prevent instant reinfection when they're reconstituted. They will also contain data that's transitionally corrupted or inaccessible, requiring that valid data be selected from multiple historical images and composed into a single consistent dataset. Finally, whatever the initial vector for attack through which the malware originally entered the system must be blocked and sealed (changing passwords, strengthening firewall rules, etc.)

Keep in mind that even this remediation process is itself subject to ransomware attack; since the original system was vulnerable, at least initially its reconstituted form is subject to the same type of attack, until the attack vector is discovered and closed. Additionally, it needs to be protected against reinfection via channels the attackers may have discovered in the initial attack (compromised credentials, other discovered system vulnerabilities, etc.) To assure that the system is functional, has a valid and desirable dataset, and is not subject to reattack it must be extensively tested. And this process, too is subject to attack. Therefore, it's critical to perform both the reconstitution of system and the testing of a candidate system in a sealed and protected environment.

Persistent's CyRev product addresses these side-ranging aspects of ransomware incursion and the process of protecting, discovering an attack using ML algorithms, reconstruction of impacted images, storing them in the vault and testing them in the air gapped environment . The following sections describe how the solution is deployed and operates in your infrastructure.

Deployment Scenario

This section describes the CyRev integration with your organization's infrastructure:



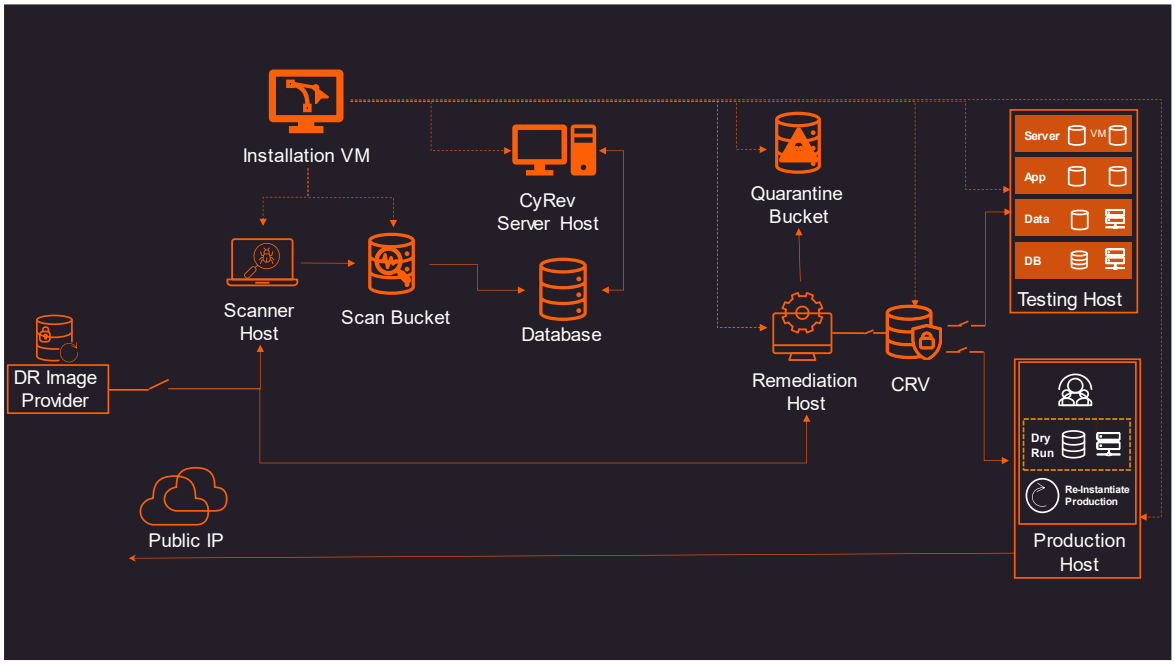
To protect a system and react to ransomware attacks CyRev requires access to images of the system on an ongoing basis, in order to scan them for attacks, allow for the creation of cleaned images, etc. CyRev gains access to these system images through a Trusted Image Provider (**TIP**); in this case, Google Cloud Backup and Disaster Recovery GCBDR.

In a typical GCBDR deployment GCBDR components are deployed in your production environment to capture system images and store them in cloud storage. CyRev requires access to two types of GCBDR components: the Backup servers that provide access to captured images, and the Management Console. The CyRev solution is deployed in same GCP project and is connected to the necessary GCBDR components to gain access to images for the systems/applications being protected.

Below section helps you understand the architecture of CyRev, and how it helps to recover from ransomware attack.

CyRev Architecture

CyRev is a Cyber resilience product that protects your organization from various cyber threats mainly the Ransomware Attacks. Using CyRev, you can create a complete Cyber Resilience Recovery Plan to scan the DR images as they are created. CyRev's architecture consists of several components that work together to discover and react to ransomware attacks.



The key components of the architecture are:

- **DR Image Provider**: CyRev connects with the Google Cloud Backup and Disaster Recovery (GCBDR) Management Console and GCBDR Backup server in order to gain access to system images of systems (applications) being protected. GCBDR captures the images in the production environment and stores them in the cloud environment. (For more information refer to the Backup and Disaster Recovery Documentation.) CyRev accesses and uses the DR images stored by GCBDR to perform the scanning and remediation operations.
- **CyRev Installation VM:** The CyRev is deployed in your environment from GCP marketplace launcher. On the initial deployment of CyRev an *Installation VM* is deployed in the CyRev project. This component controls creation, management, and deletion of other CyRev resources that are not created as a part of initial deployment, such as the Test Management Host, Production Management Host, etc. The Installation VM is accessible via RDP from your computer.
- **CyRev Scanner Host**: The CyRev Scanner is a cyber resilience technology developed by Persistent Systems Ltd to scan the DR images to safeguard the

organization from various cyber threats. The Scanner host scans system images to detect anomalies and threats and raises alerts based on threats it detects.

- **CyRev Scan Bucket**: The Scan Bucket is the repository of the results of image scans. The scanning results from the Scanner Host are stored in the scan bucket, and other components like the CyRev database access it.
- **CyRev Server Host**: The CyRev Server Host presents the CyRev web interface which helps you to quickly identify the threats and attacks on the candidate images and review the changes in them using a dashboard that allows fast review and action on hundreds of candidate images.
- **CyRev Remediation Host**: The remediation host performs the job of constructing clean images after an attack is detected. Using instructions on what to do to with corrupted files (For example: delete or quarantine them) and as result it produces a clean data image.
- **CyRev Quarantine Bucket**: Any files determined to be infected in system images during the remediation process are stored in the Quarantine Bucket to isolate them. These files will be available for forensic purposes.
- **Cyber Resilience Vault (CRV)**: The CRV is an air gapped, immutable storage system used to store and protect cleaned images, which are used for testing and production.
- **Test and Production Environments**: When candidate images are produced CyRev will deploy an isolated testing environment where the cleaned images can be tested and verified. Once cleaned images have passed the testing phase, they can be deployed at scale into a new production environment.
- **Cassandra Database (DB)**: The database stores scan information collected from scanner host for analysis and sends the updated information to UI Dashboard.

These CyRev components orchestrate the Cyber Recovery workflow at scale and provides the opportunity to rapidly react and recover from a ransomware attack.

Getting Started

Before using CyRev, it must be deployed into your environment using the instructions in the *CyRev Getting Started Guide*.

The following sections describe how to access and use the various CyRev components to configure protection against ransomware attacks upon your systems and applications, and how to react against any detected attacks.

- The CyRev UI presents you with information regarding ongoing and completed scans and any detected threats and anomalies.
- The CyRev Scanner performs the scans on images, looking for indications of an imminent or ongoing attack.
- CyRev Remediation involves creating a clean system/application image to replace a compromised one.
- Once a remediated image is created, you can launch a testing environment in order to verify the image is clean, complete, and fully functional as a replacement to the compromised one.
- After testing to verify you have a clean and functional image CyRev assists in Launching a Production environment and resume normal operations.

CyRev UI

The CyRev UI is the main interface for monitoring ongoing CyRev operations. The main component is the CyRev dashboard which shows the applications and systems being protected; monitors the status of ongoing and completed image scans; presents alerts about detected anomalies and threats; and allows you to drill into the details of the files that may contain malware threats.

Accessing the CyRev Dashboard

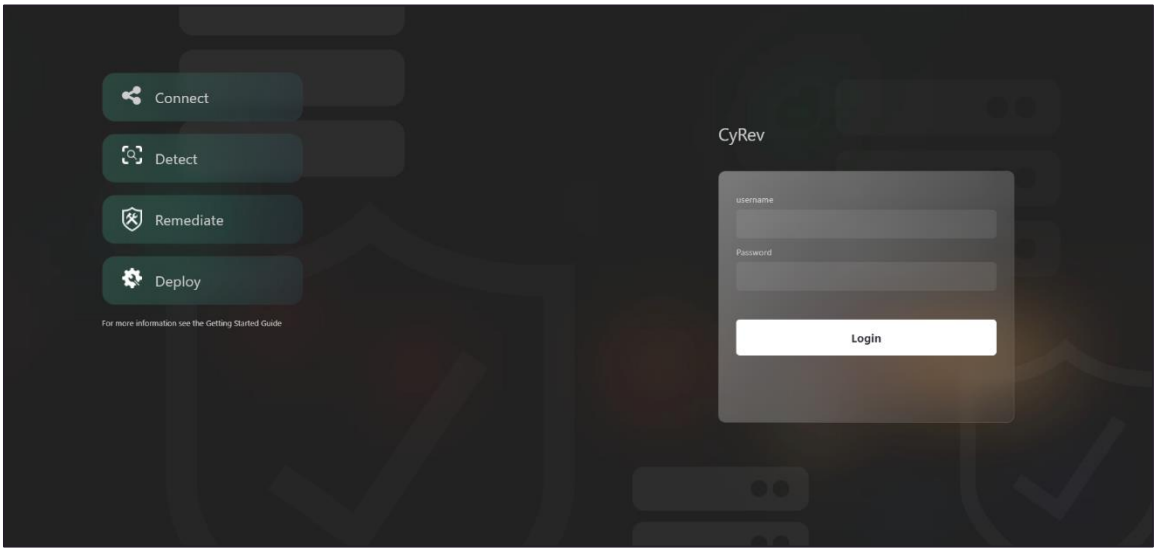
To access the CyRev UI you must launch a browser on the CyRev Server Host to connect to the dashboard. To do this you will use RDP to create a desktop connection to the CyRev server.

1. Connect to the CyRev Server Host from your desktop via Remote Desktop Protocol (RDP) using its cloud IP address.
   (**Note**: You need to create VPC peering between Jump Serve/Bastion Host and CyRev Sever Host if you have an internal IP deployment.)

2. Open the Chrome browser on the CyRev Server and connect to the UI using the following URL:

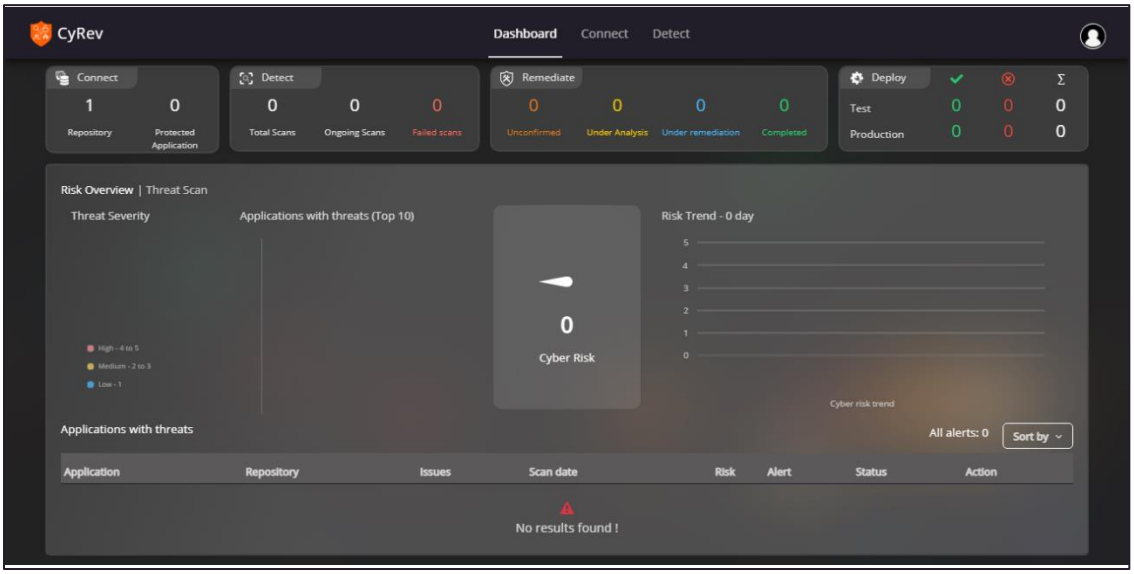
`https://127.0.0.1:3001`

3. This will show the CyRev login screen:



The first time you access the CyRev UI you need to use the default login credentials created in CyRev Server Host Metadata.

After logging in you will are presented with the CyRev Dashboard. Upon your first login the dashboard data will be unpopulated:



Later sections will describe how to configure applications for protection, after which the dashboard will be populated.

The following sections give a tour of the pages and components of the CyRev UI.

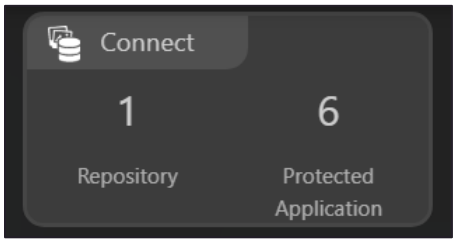
Dashboard Cards

The CyRev Dashboard has good amount of information about your systems and graphical information to quickly identify the issues that need your attention. It is comprised of a series of cards to succinctly present related information.

Connect

The *Connect* card indicates CyRev's connectivity with the source(s) of application images. It presents the following parameters:

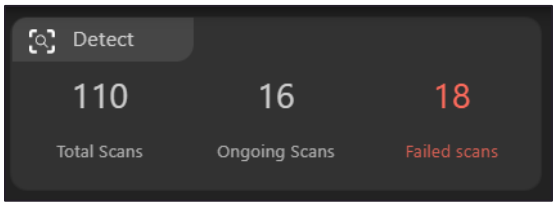
- *Repositories*: Repositories are the image store that provides the DR images to CyRev. The count in connect card shows the total number of GCBDR Backup servers that have been registered with CyRev.
- *Protected Applications*: The protected application are the applications are scanned and protected by CyRev. The count shows the total number of applications configured for protection by CyRev.



Detect

The *Detect* card shows information about the image scanning process. The card contains following parameters:

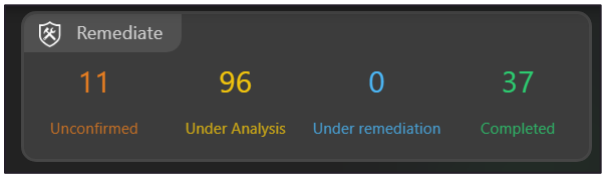
- *Total scans*: Shows the count for the total images scanned by CyRev.
- *Ongoing scans*: Shows the number of ongoing images scans.
- *Failed scans*: Shows the count for the number of failures encountered while performing image scans.



Remediate

The *Remediate* card shows information about the status of threats and anomalies discovered by CyRev. It displays the following parameters:

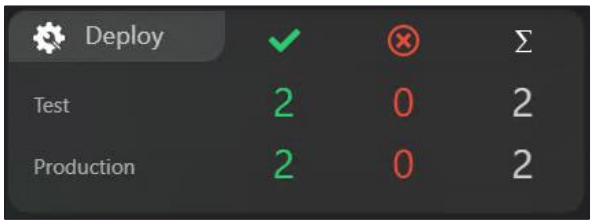
- *Unconfirmed*: Displays the count for the unconfirmed issues detected during application scans by CyRev.
- *Under Analysis*: Shows the count for the issues that are under your analysis.
- *Under Remediation*: Displays the count of issues that are under remediation.
- *Completed*: Displays the count of issues whose remediation is complete.”.



Deploy

The *deploy* card shows information about the recovered test and production VM deployments. Card contains the following parameters:

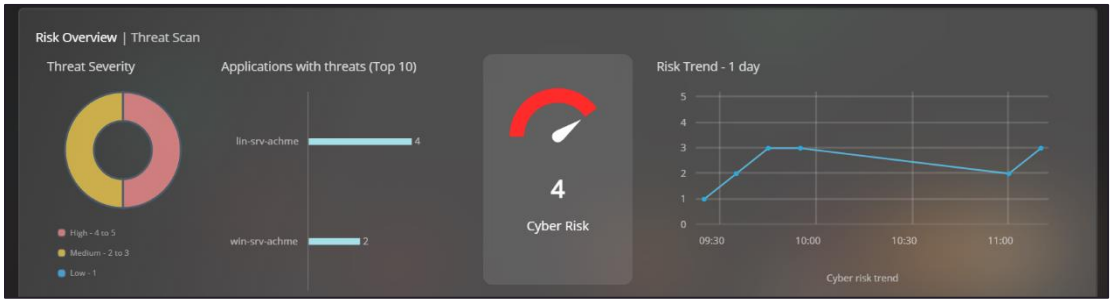
- *Test*: Displays the number of passed and failed Test VM deployments.
- *Production*: Displays the number of passed and failed Production VM deployments.



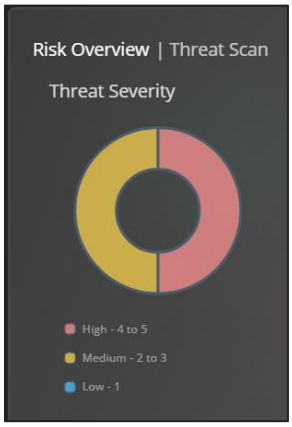
Risks Overview

The **Risks Overview** section has 3 card which gives a graphical representation of threats and anomalies discovered by CyRev. It gives a perspective on what types of threats have been discovered, a view of the most severe threats and the trend of threats over time.

In the CyRev UI, anomalies discovered during scanning (such as high-entropy files, indicating a possible encryption wave) are classified as *threats*. Scanned images are assigned a *Cyber Risk* value between 1 and 5 based on the number of files indicating a potential threat as a proportion of all files evaluated in an image (a single scan evaluates any files that were modified or newly added since the prior scanned image.) A Cyber Risk value of 1 or two is considered a *Low* risk value; 3-4 is deemed *Medium* risk and 5 is *High*.

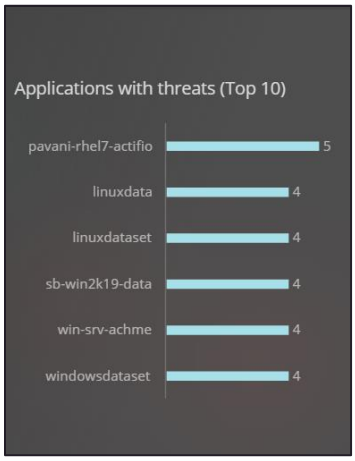


Threat Severity Chart: The Severity Chart shows the relative proportions of the risk severity of discovered threats . Low risk (1-2) is shown in blue, medium risk (3-4) in yellow and high (5) in red, and the pie chart shows the relative proportion and precise percentages of each level.



Applications with threat (Top 10)

The *Applications with threat* graph show the list of top 10 applications with the highest Cyber Risk values.

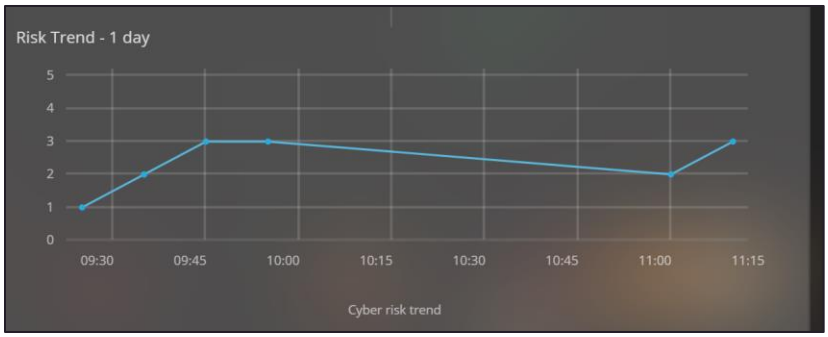


Cyber Risk

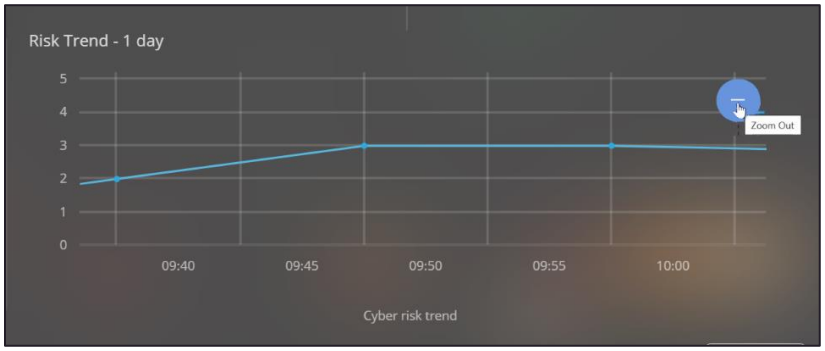
The *Cyber Risk* Meter shows the overall cyber risk score aggregated over all the images scanned by the CyRev system. The color of the meter varies depending on the risk score. For scores 1, it is green, for scores 2 and 3, it is yellow and for scores 4 and 5, it is red. The lowest risk score is 1 and the highest is 5.

Risk Trend

The *Cyber Risk Trend* graph shows the progression of the Cyber Risk score over time. You can see the progress of threats affecting all systems being scanned. The graph now displays the changes in risk on a per-minute basis. You can zoom in the graph by selecting the area as shown in below screenshot.

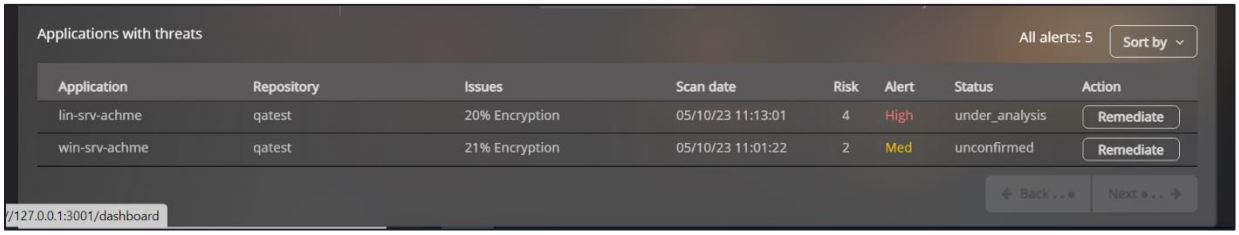


You can restore view by clicking Zoom Out button.



Application with threat Table

At the bottom of the dashboard is the *Application with Threat* table that lists applications with threats that require your attention. Each row represents an issue detected in a single image of an application.



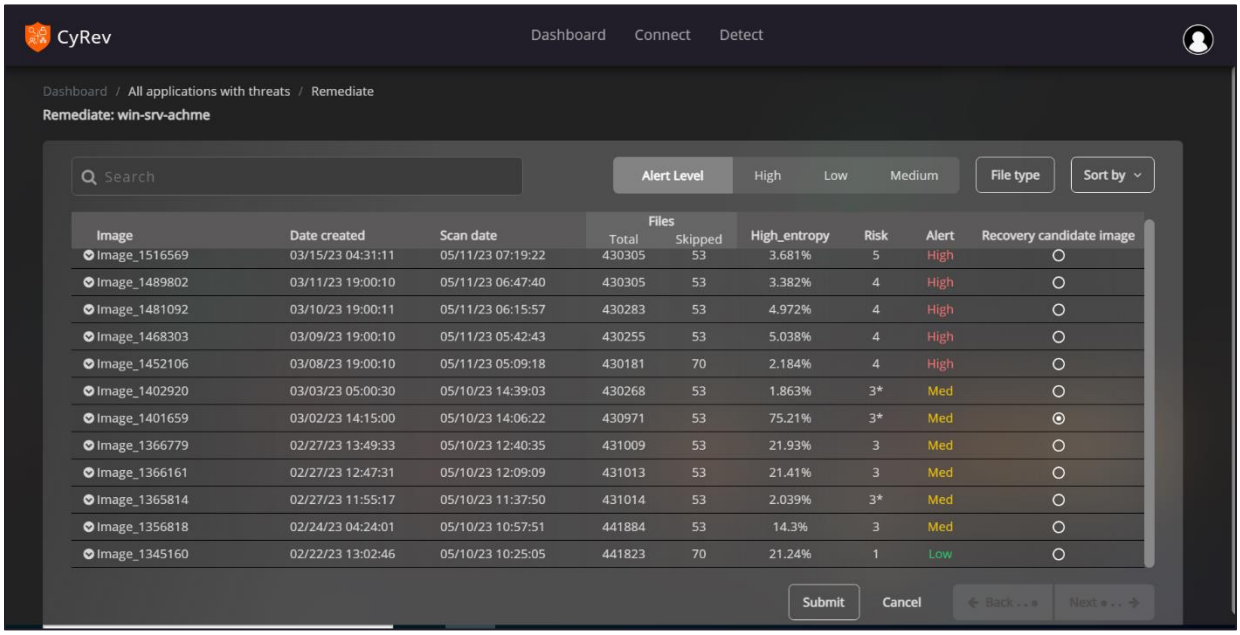
For each image the table displays:

- *Application*: The name of the application whose image is listed.
- *Repository*: This column contains the name of the repository to which this application belongs.
- *Issues*: The column contains the percentage of files encrypted for each application.
- *Scan date*: The date and time the image was scanned by CyRev.
- *Risk*: The Cyber Risk value calculated for the image (1-lowest to 5- Highest).
- *Alert*: The alert level of the application calculated based on cyber risk value (1-2 = Low, 2-3 = Med, 5 = High)
- *Status*: The status of the image based on remediation status of the application.
- *Action*: This column contains the *Remediate* button; click it to get more detail about the application and its images. This will open the *Remediate Page*.

Remediate Page

Click on **Remediate** button in the Action column of the **Application with threat** table to open the *Remediate page*. On the Remediate page you can see the details about the

results of scans of images of an application scanned by CyRev.



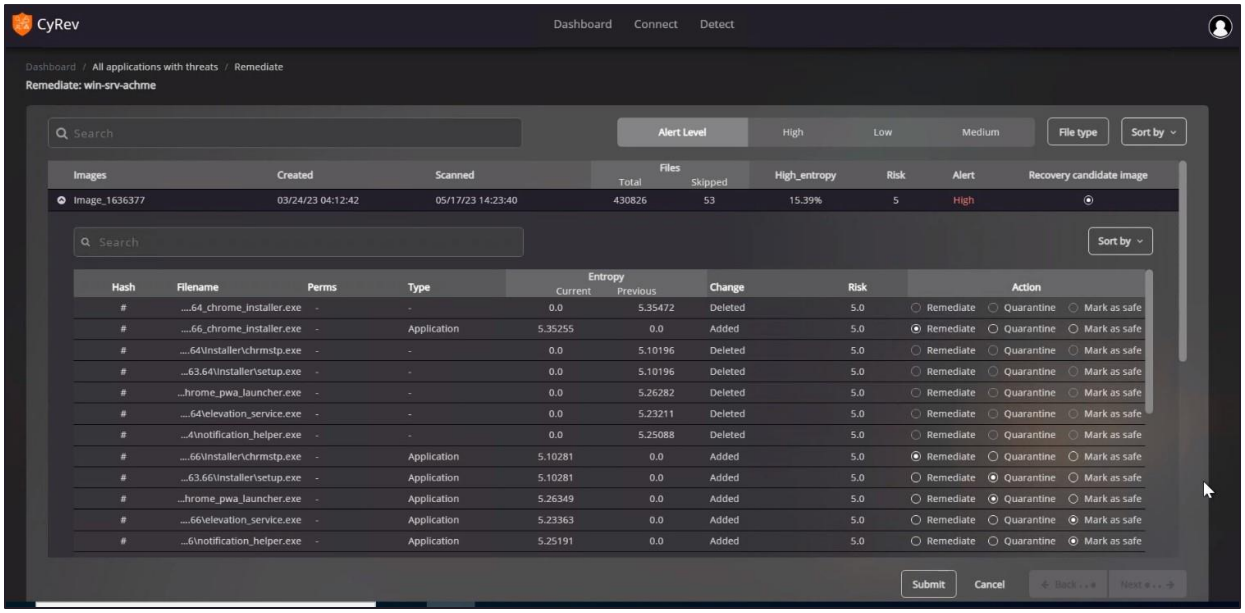
For each image you will find the following details:

- *Images*: The name of the application image.
- *Date Created*: The date and time of the image capture.
- *Scan Date*: The date and time of scan.
- *Files*: This column is divided into two parts that are Total and Skipped. Total column lists the number of files that are available on the image. Skipped column lists the number of skipped files during the scan.
- *High Entropy:* This column lists the percentage of encrypted files on the image versus the number of files scanned files (compared to previous scan).
- *Risk*: The Cyber risk calculated for the image (1-lowest & 5- Highest).
- *Alert*: The severity of the Cyber Risk for the application image (Low=1-2, Medium=3-4, High=5).
- *Recovery Candidate Image* : This column contains the *Radio* button to select recovery candidate image. You can explore more details about the files by clicking on drop-down arrow. This opens the list of *File Change page*.

The list of files in this page can be sorted by clicking on High, Low and Medium filters also you can sort the images by clicking on Sort by or you can search for the required file by using search option. You can also select the File Type filter to sort the type of files across multiple images.

Files Detail Table

Clicking the image name row you can access details about the files associated with each image that were scanned and determined to be of concern.



The File Changes page shows the following information:

- Hash: This column contains the hash code for the files to allow faster search of the files across and within the image.
- *Filename*: This column lists the detailed file path and name of the file that was determined to be a potential threat.
- *Perms*: The file permission that CyRev has for the file.
- *Type*: This column lists the type of that file, and you can sort the files based on the file type from the drop-down.
- *Entropy Current*: The entropy value measures the randomness of the data in a file and is used to determine whether a file is encrypted or being encrypted. Entropy values range from 0 to 8, where 0 is the least likelihood of encryption and 8 is the highest. The entropy the column is divided into 2 parts that are *Current* and *Previous.* The Current entropy column shows the entropy value for the file during the current scan that is performed on the images. Similarly *Previous* column lists the entropy calculated for the file in the previously scan.
- *Change:* This column lists the change type for files that are updated between the subsequent scan The status is Updated, Added, Encrypted and Deleted.
- *Risk:* This column lists the risk for each file ranging from 1-lowsest to 5- highest.

- *Action:* This column has three selections Remediate, Quarantine, and Mark as safe. You can act based on your analysis.

Refer below section to perform the scanning, remediation, and recovery operations. The list of files in this page can be sorted by clicking on Sort by or you can search for the required file by using search option. Use Back and Next button to navigate across the pages.

CyRev Commands

Most of the operational interfaces for CyRev are commands executed on the appropriate CyRev component (Remediation Host, Test Host etc.) (In future releases this functionality will be driven from the UI.)

CyRev command scripts generally perform a single operation (creating remediation images, creating test host, etc.) These operations in turn are related to an application and its images, and in turn to the connection to the image provider (GCBDR) from which CyRev gets the application information and application images. The commands also need information about the operating environment (such as the CyRev and GCBDR projects, network information, etc.)

The CyRev commands get the necessary information about the application, images and environment from a configuration file which is supplied as an argument when the command is executed. It is probably most convenient to have a configuration file for each application being protected.

Command Configuration Files

To perform some CyRev operations such as Testing A Candidate Image or Deploying Image in production environment it's necessary to run commands on the appropriate CyRev Host. To do this you will first need to connect to the appropriate host via RDP from your computer, and then launch a command line or PowerShell prompt. Commands are then executing at the command line.

CyRev Commands are all controlled by configuration files that contain parameters to define various aspects of the command's operation, such as how to connect to the image provider, which application and image should be operated upon, etc. The specifics of each command's configuration file are outlined in the sections pertinent to the operation later in this document.

You will specify the name of the configuration file you want to use as part of the command line when you perform a given operation. If you are protecting multiple applications/systems, you will likely wish to keep a set of configuration files for each one; this saves having to modify the same configuration file multiple times when you switch protected applications/hosts.

Many of the parameters in the configuration files are related to the deployment of CyRev, the deployment of the GCBDR image provider and the GCP environment in general. The following table summarizes the environmental and deployment information you might need when executing CyRev commands; it's recommended that you fill it out so that you have the information handy as you perform various operations:

CyRev Components		
CyRev Project	CyRev Custom Service account email Address	Cust SA Email ID:
Installation VM	The name/IP address, Access credentials.	ID/IP: Login:
CyRev Server Host	The name/IP Address, VPC and credentials for the CyRev Server Host.	ID/IP: Login:
Test Management Host	The name/IP Address, VPC and credentials for the CyRev Test Management Host.	ID/IP: VPC: Login:
Application Test Host	The name/IP Address, VPC and credentials for the CyRev Application Test Host.	ID/IP: VPC: Login:
Production Deployment Host	The name/IP Address, VPC and credentials for the CyRev Production Deployment Host.	ID/IP: VPC: Login:
Application Production Host	The name/IP Address, VPC and credentials for the CyRev Application Production Host.	ID/IP: VPC: Login:
GCBDR Parameters		
API URL	GCBDR Management Console API URL used by CyRev to access DR images	URL
OAuth 2.0 client ID	Client ID of GCBDR	ID

You will also find a spreadsheet version of this table here: CyRev Deployment Information.

Using CyRev

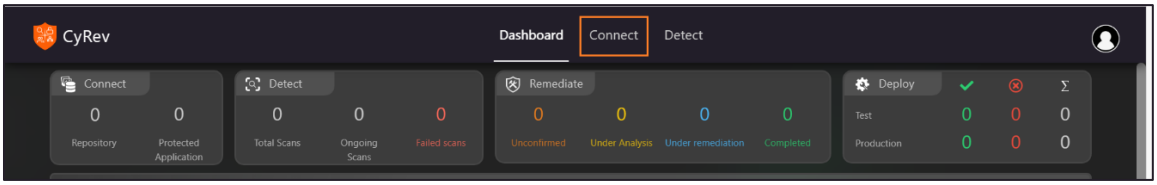
Using CyRev naturally falls into two phases of operation:

- *Steady State Operations* are performed on an ongoing basis in order to establish protection of designated applications and systems. This involves scanning images for those applications/systems, detecting, and reporting anomalies and analyzing and classifying issues raised in the scanned images.
- *Remediation Operations* are undertaken when a replacement for a protected application/system needs to be created. This may be in reaction to a discovered ransomware attack, as a test of the ability to handle an attack, etc. This includes designing, creating, testing and deploying a sanitized image of the original protected application/system and deploying it for use.

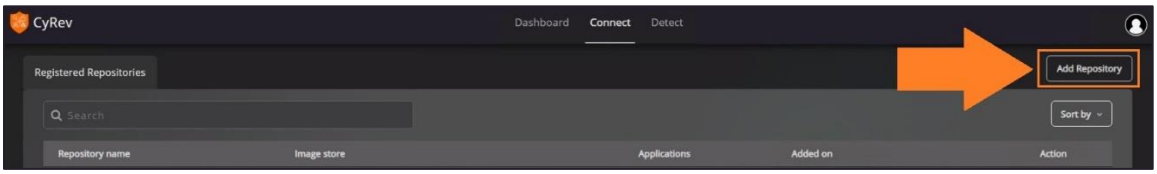
Connecting Image Provider

After the successful account configuration and you need to connect CyRev to image provider to access the images. In order to access application information and images from the GCBDR image provider the CyRev need to have credentials for access to the GCBDR management plane and image store. Follow the steps to connect image provider:

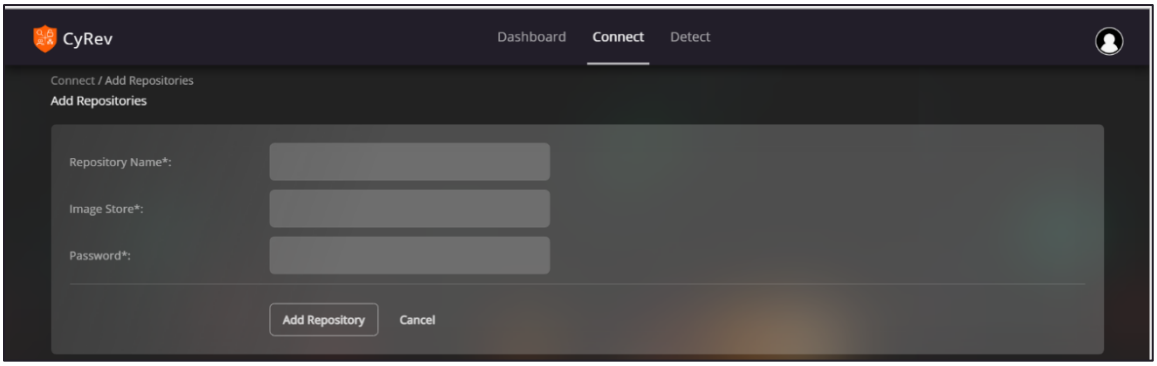
1. Connect to the CyRev Server using RDP from your computer.
   (**Note**: You need to create VPC peering between Jump Serve/Bastion Host and CyRev Sever Host if you have an internal IP deployment.)
2. Login to the CyRev Dashboard using username and password.
3. Click **Connect** on top pane.



4. Click **Add Repositories** on left pane. The Add Repositories page opens.



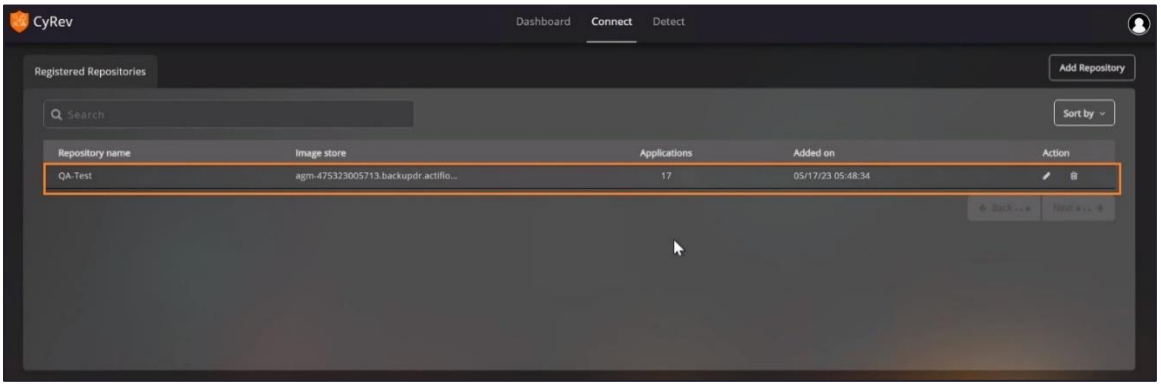
5. Configure the fields as described in below table:



Parameter	Description	Example
Repository Name	Enter name of the GCBDR Repository of your choice.	`Project Repository`
Image Store	Enter the GCPDR Management console API URL. You can obtain it from GCBDR	`Agm-152650448883.backupdr.actfiogo.com`

	cloud console page. **Exclude the https:// from start and /actifio from URL**.	
Password	Enter GCBDR OAuth 2.0 client ID as a password. You can obtain it from GCBDR cloud console page.	`48652520581547-dfsdf017dasdsacsd1ds.apps.googleusercontent.com`

6. Click **Add Repository** to register repository with CyRev. You will receive notification after successful addition of the repository.
7. You can see newly added repository in the list.

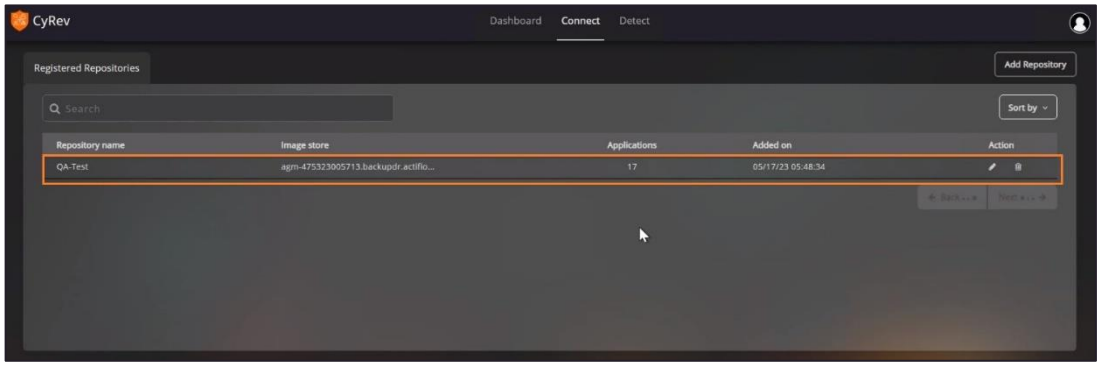


If repositories details need to be updated, you can update the repository details as indicated in the Edit Image Provider Repository Details section.

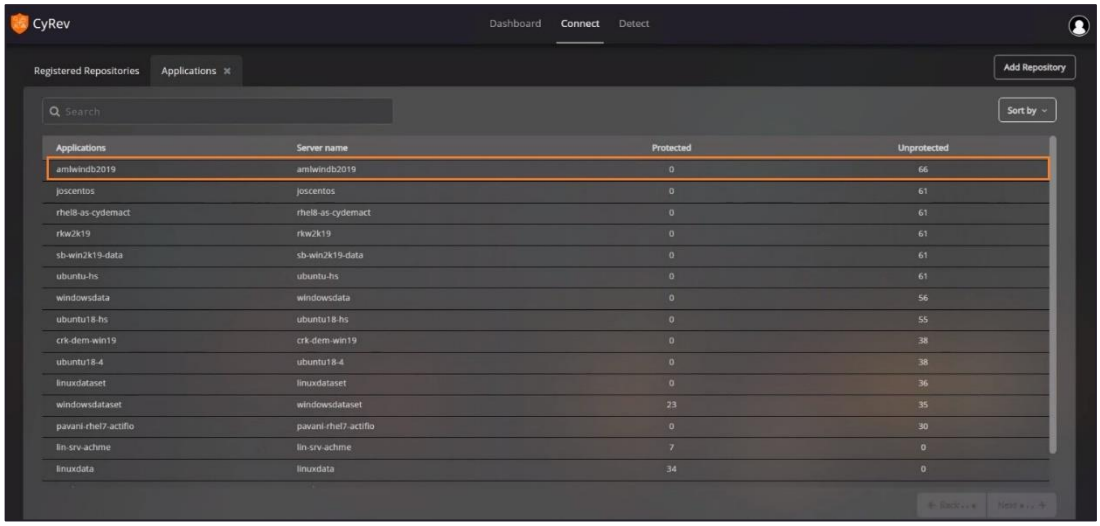
Image Discovery

After successfully registering a repository, CyRev automatically initiates the image discovery operation. In which CyRev lists the applications that are backed up on GCBDR and the images that are available for each application. Follow the steps to verify image discovery operation:

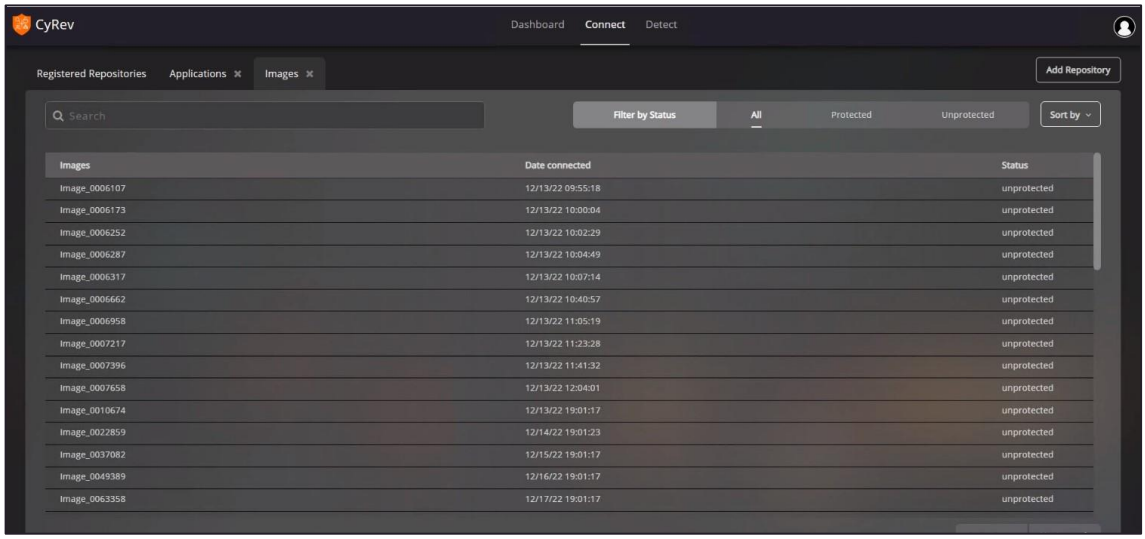
1. Log into Dashboard and click **Connect**.
2. Click the registered repository row to access the list of registered applications with repository.



3. You can see list of registered applications with the repository. Click the application from the list.



4. You will see the list of images that are available for the application:



Scanning Application and Images

To get information about threats to applications, CyRev scans the application images made available by the GCBDR image provider. These operations are performed on the CyRev Scanner host, which attaches the images from the image provider, scans the files in the images and stores the results for further analysis and display via the CyRev UI.

To perform a scan, it's necessary to first discover the list of images available for the application/application being protected. You can then choose images to scan for threats, which you can then evaluate using the CyRev UI.

This cycle of discovering newly available images and then scanning them should be regularly performed. This is the "steady state" of protection using CyRev: scanning images and evaluating metrics to prepare clean images for possible future recoveries and to discover attacks as early as possible.

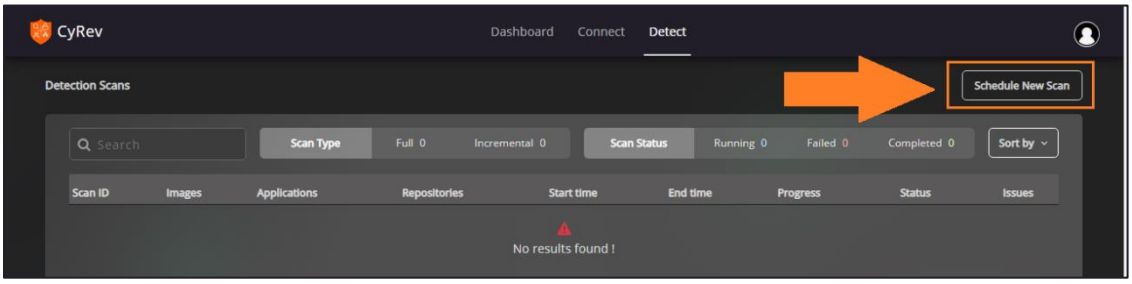
The steps of discovering and scanning images are performed by scripts that are run on the CyRev Scanner Host. The discovery/scan operation will be scheduled and/or triggered automatically when images become available through the trusted image provider.

CyRev provides two scanning options for the application form repository. You can either scan the entire application or commence scanning from a specific image. Below section describes the detail steps to scan an application and images.

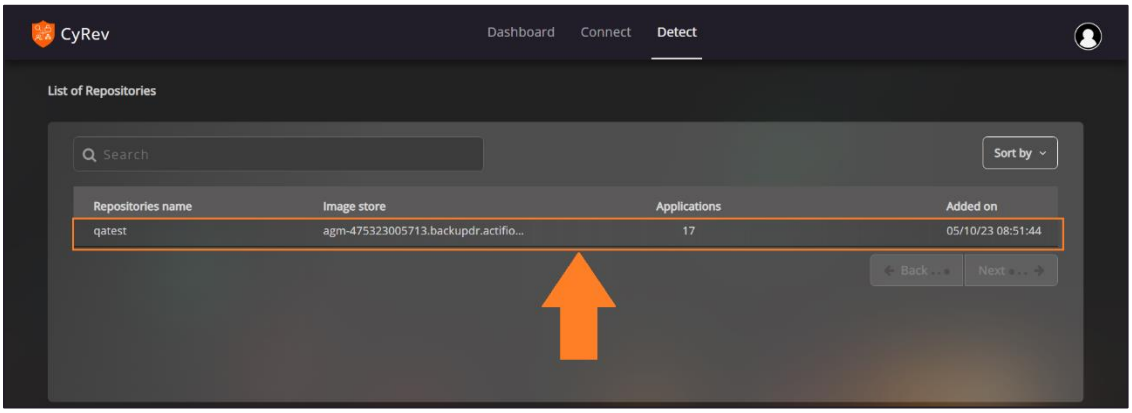
Scanning an application

You can scan the application Image for the UI by following the below steps:

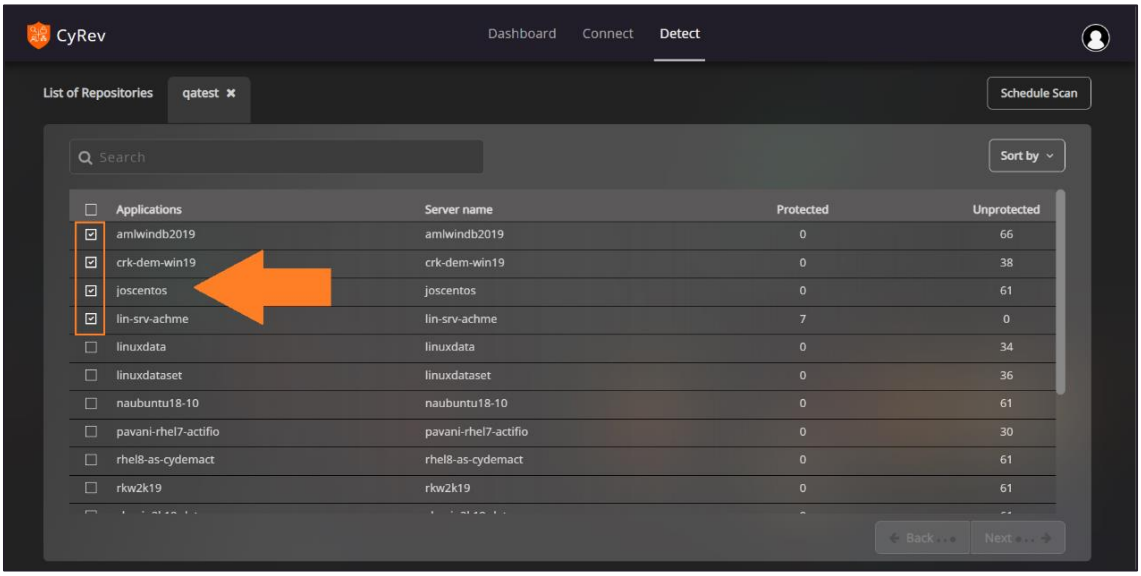
1. Click the **Detect** page from main menu.
2. Click **Schedule New Scan** button.



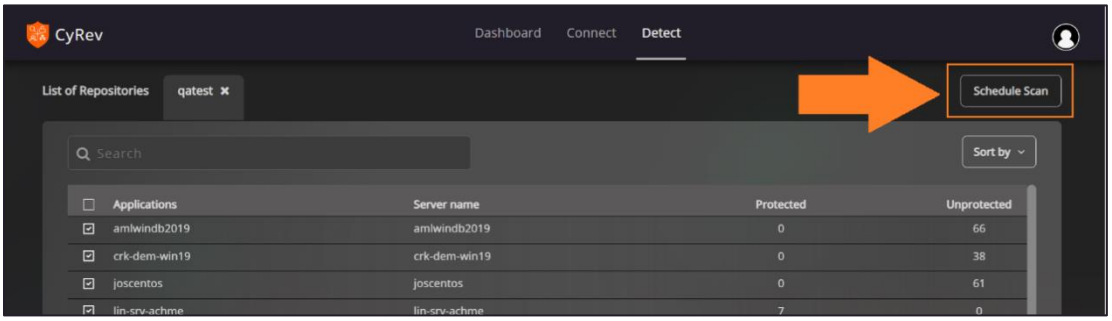
3. Select your repository.



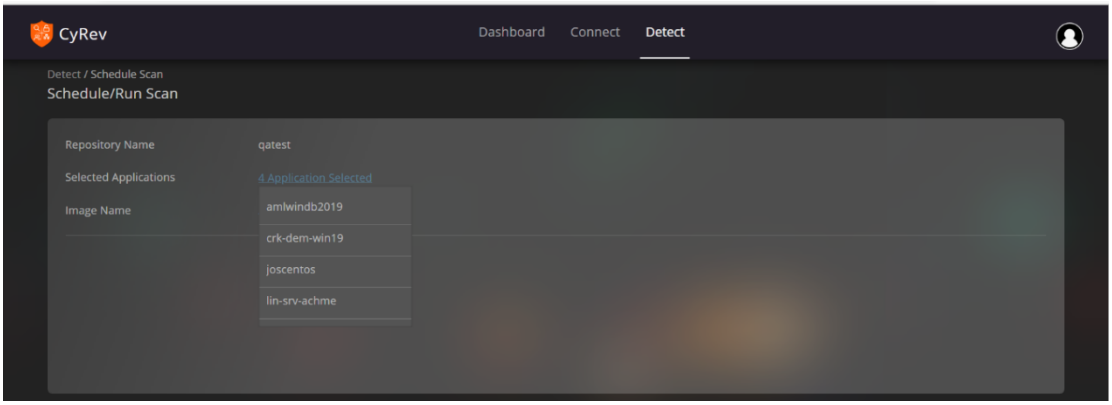
4. Check the box and select the application that you want to scan. (You can also select the multiple application from the available list.)



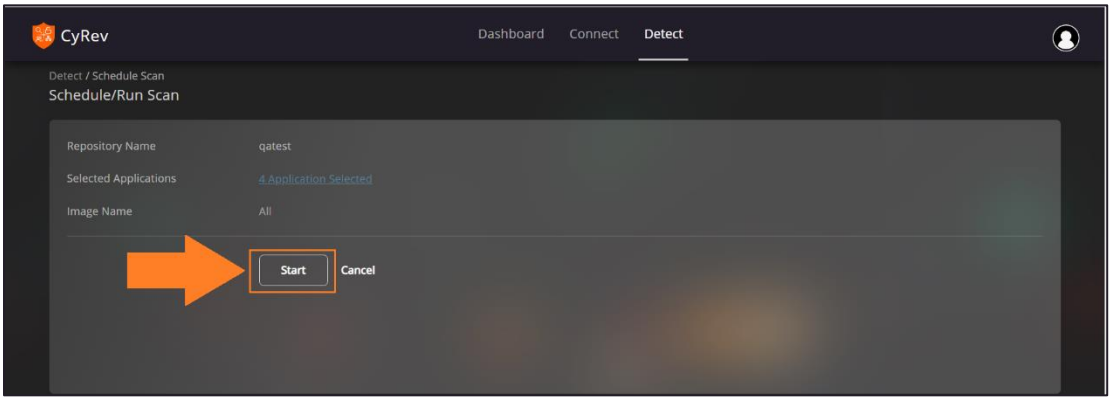
5. Select **Schedule Scan** button. Run scan page opens.



6. You see the list and number of selected applications by clicking on the Application Selected.



7. Click **Start** to initiate the scanning for selected applications.

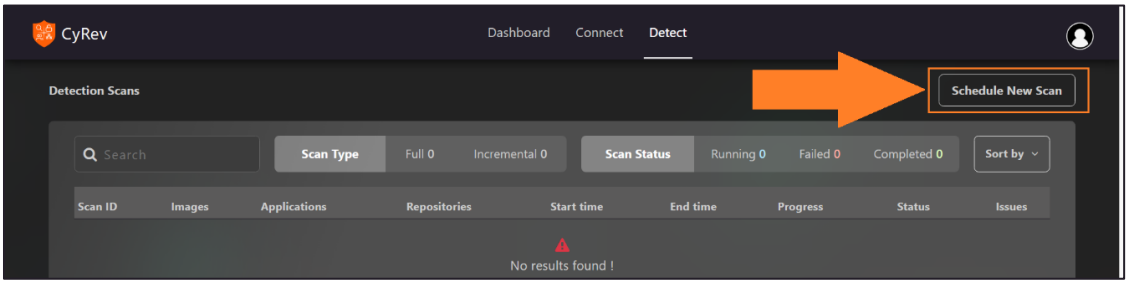


The section below will guide you in scanning the images for a specific application, as you can do with other applications.

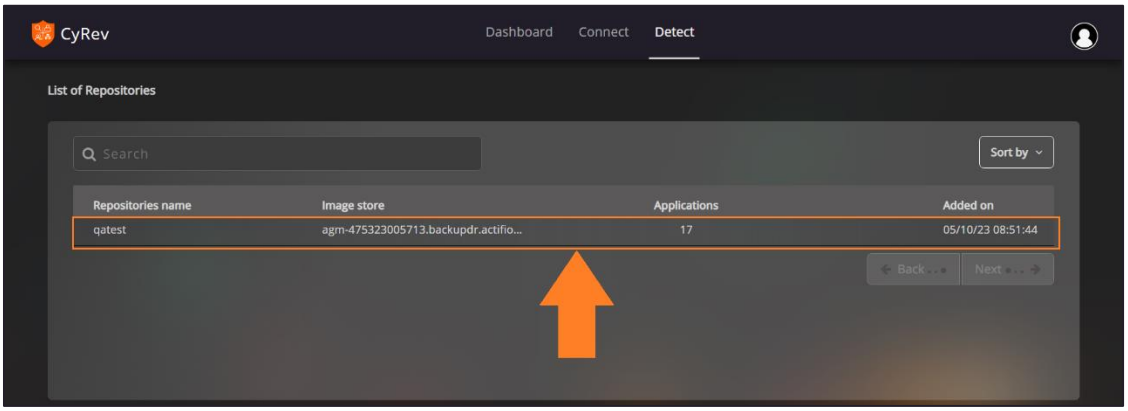
Scanning an Image

To initiate a scan for a specific image in the application, please follow the steps below:

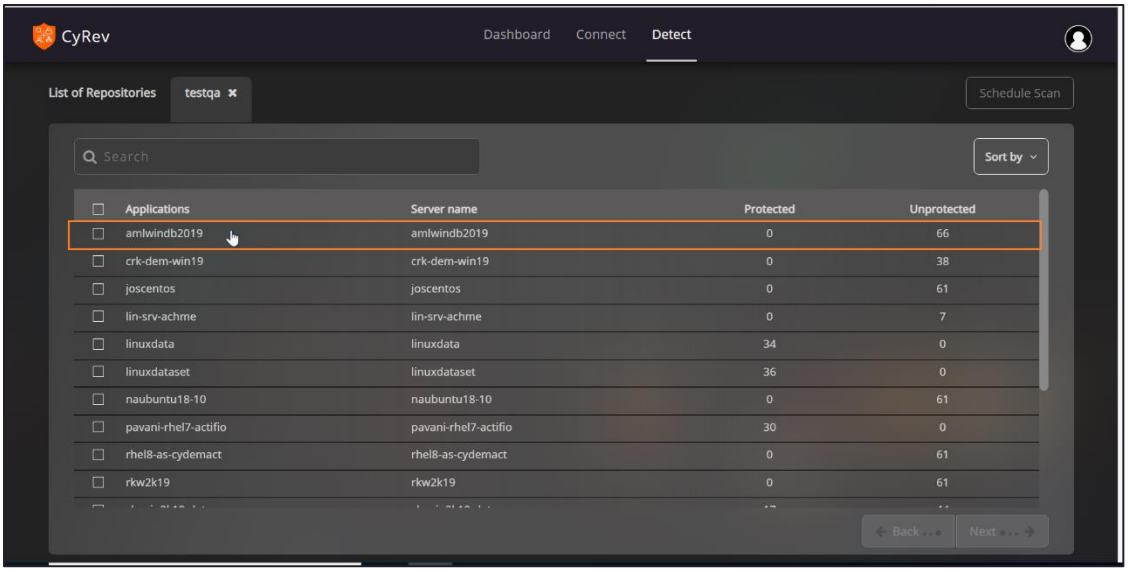
1. Click the **Detect** page from main menu.
2. Click **Schedule New Scan** button.



3. Select your repository.

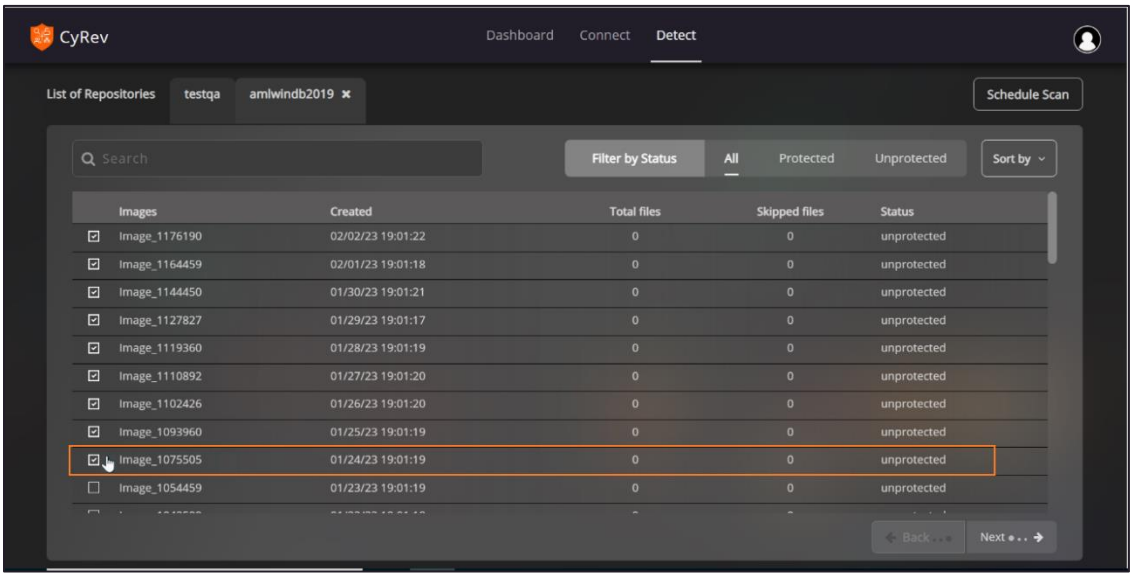


4. Select the application that you want to scan.

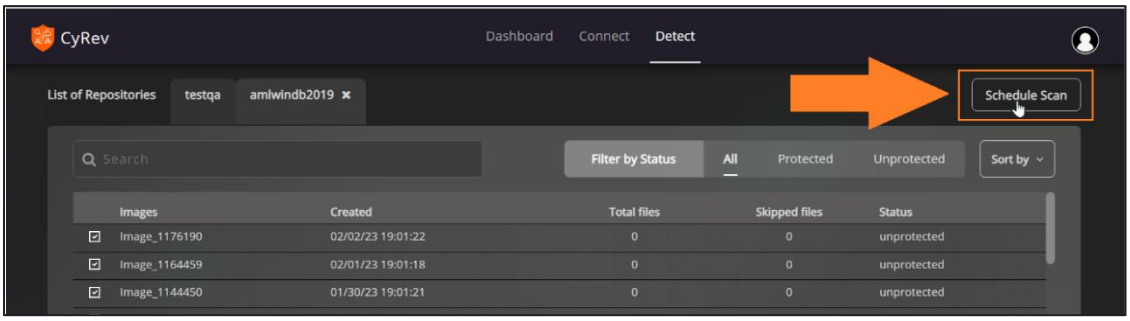


5. Choose the image from which you want to start scan from the list. You can use filters to check the protected and unprotected images from the list.

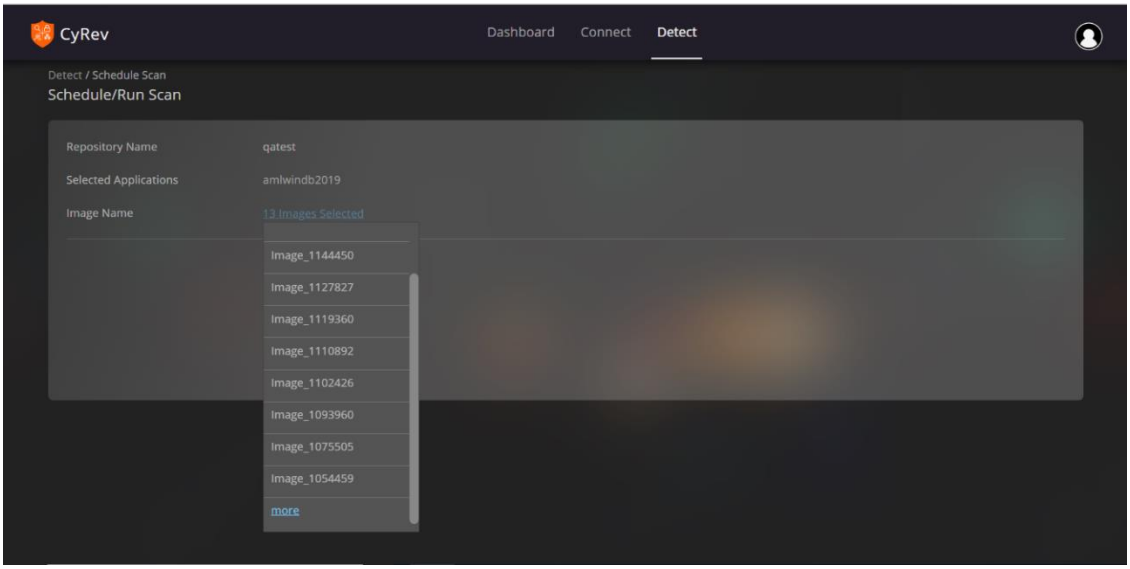
(**Note**: The first image selected from the list considered as base image for analysis and from that image onward all images will be scanned on the incremental basis).



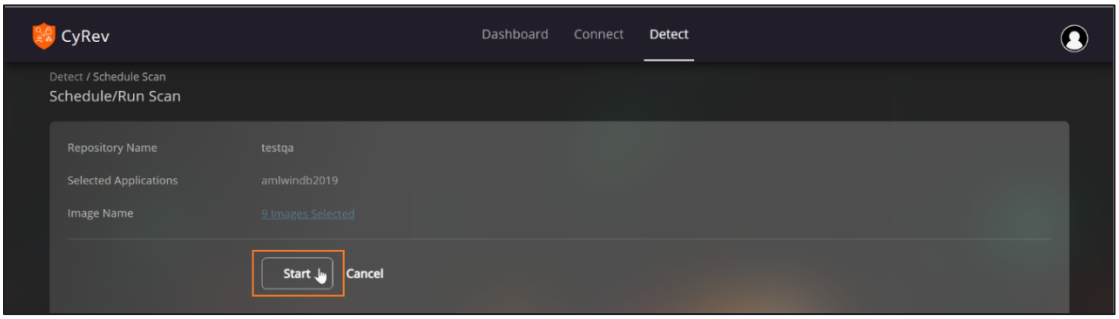
6. Click **Schedule Scan**.



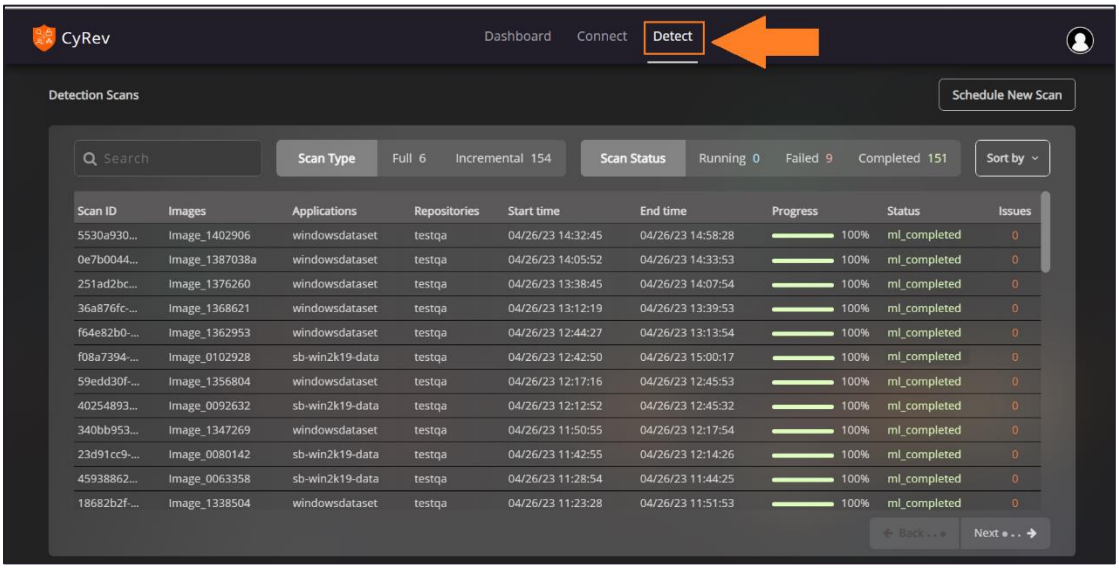
7. You can see the summary of the images that are selected for scanning by clicking on number of images selected.



8. Click **Start** to submit the scan request..



9. You can see the results of the scan by navigating to Detect page.



The UI will show progress as the scan executes. The results are stored in the Scan Bucket for later analysis.

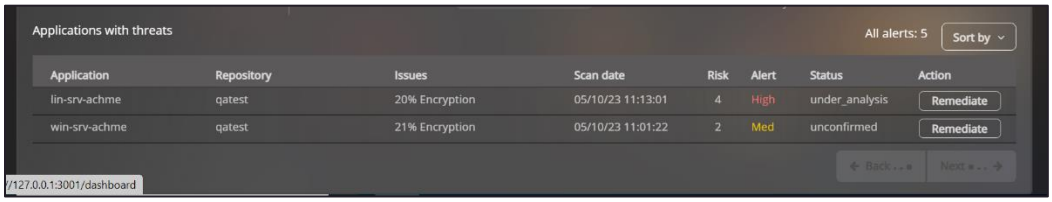
Scan Results

The results for each application scan are stored in the CyRev Scan Bucket and Database. These results are used for the analysis purpose and are accessed on the CyRev Dashboard. The next section covers viewing and analyzing scan results.

Analysis and Detection

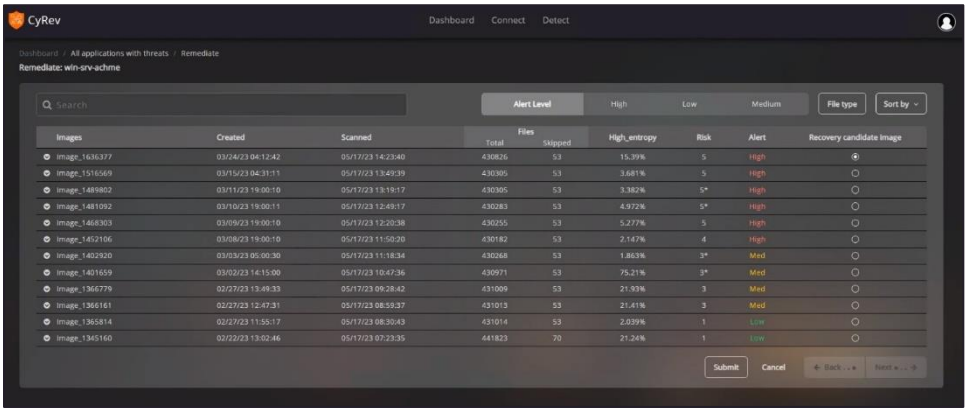
The scanner CyRev calculates the entropy for the files in application images and tracks the change in entropy across images. This entropy value and the change of a file's entropy over time is the basis on which CyRev evaluates threats.

Once CyRev analyzes images it evaluates the results and determines if there are any potential anomalies or threats to the application/system. These threats are displayed in the CyRev UI, in the **Application with threat** table on the Dashboard:



The Application with threat table displays the application which have anomalies that may indicate malware or ransomware present or in operation. Which means an unusual number of encrypted files were discovered. It assigns a **Cyber Risk** and **Alert level** based on its findings, based on the number of at-risk files are discovered as a percentage of the encrypted files that have changed from the prior image scan.

To investigate the issues that CyRev raises, click the **Remediate** button for the application or image you wish to examine. This opens the **Remediate Page** of the CyRev UI:

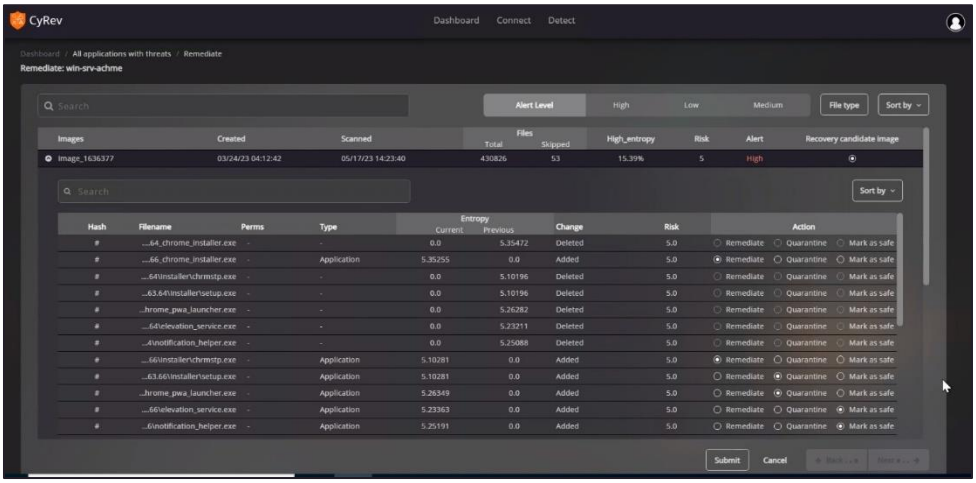


The Remediate page shows the images that have been scanned for the selected application, along with metrics about each image indicating which images seem to have a problem; this is indicated in the **Cyber Risk** and **Alert Level** values.

The most common reason for an image to be flagged as problematic is because of large number of encrypted files, files with high entropy, or a sudden rise in the number of files

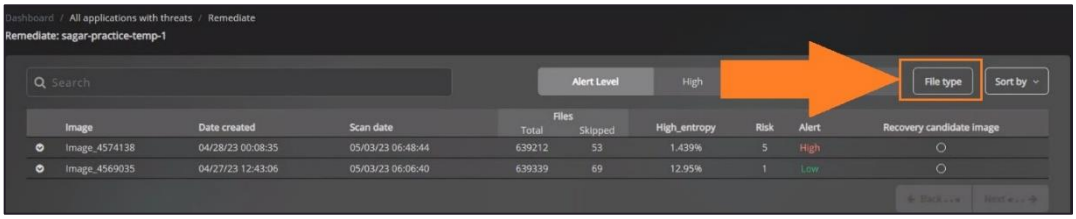
encrypted. CyRev uses an entropy metric to determine the likelihood that a file is encrypted, and you can use that value to figure out which files are potentially being raided by ransomware on the system.

To find which files are potentially under attack, click the **Image Name** for an image that shows a high risk/alert level. This brings up the **Files Detail**, a list of all the files in that image along with attributes of the file including the entropy calculated during scanning (**Entropy**) and the previous entropy calculated in that file in the previous scan operation (**Previous Entropy**). You can also check the risk assign to each file that helps you to identify the ransomware files:

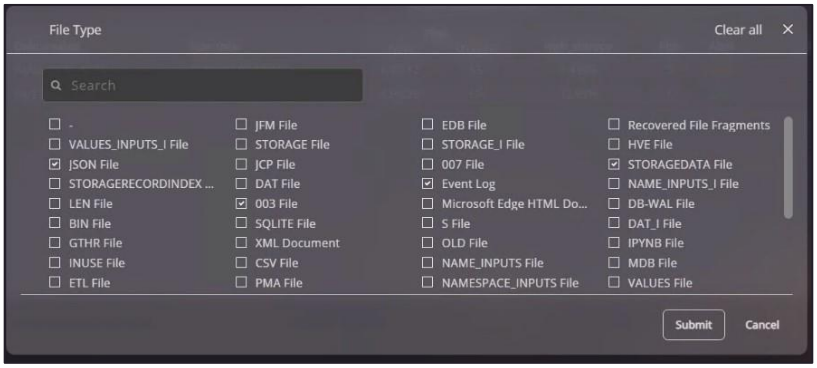


A sudden and significant increase in the entropy of a given file may indicate that it's being encrypted in place by ransomware as part of an attack. Similarly, a large number of new files with high entropy values may indicate file copies being encrypted in anticipation of replacing the originals as part of the final stage of a ransomware play.

To access a particular file from the images, users can utilize the **File type** menu button, which provides an array of filters for different file types.



By typing in the search bar, users can quickly search for the required file type. Multiple file types can be selected by marking their corresponding checkboxes. Once the desired file types are selected, clicking the **Submit** button will generate relevant search results.



If the user wishes to remove the filter, they can click the File Type button and choose the Clear all option.

Generating A Remediation Image

Once you have determined a ransomware attack has been initiated on your application/system. You can start creating a clean replacement for the original system. The idea is to create a clean system image based on a scanned image available from the image provider such that:

- Any malware files that infected the system or can perpetuate an infection have been removed or quarantined.
- Any victim files (For example: corrupted, encrypted or otherwise tainted data files) have been quarantined and replaced by known good versions.
- Any credentials that may have been compromised by the malware while it was resident on the original system have been changed or invalidated; and
- Any security holes through which the malware originally attacked the system have been closed.

CyRev assists in the Remediation process by giving access to the available history of images along with the scanning results for all the files; this allows you to select the best source image on which to base a replacement production application/system. It also allows you to identify potential malware files and victim files and quarantine them, as well as find replacement data files for the victims.

You can use this information to build a **Remediation Plan** that maps out the files to be used to compose a candidate replacement image. Using the Plan that you specify, CyRev builds a candidate replacement image – a **Remediation Image** - from the chosen application image, quarantining and deleting files as specified. This candidate image can then be further conditioned (data files replaced, credentials changed, security patches applied, etc) and tested before deploying as a replacement.

The process of deploying a replacement for the originally infected application/system is recursive: you generate a Remediation Plan and use it to create a candidate Remediation Image. This is then in turn used to create VMs to test for proper functionality, data completeness and to verify that it's no longer subject to attack. If the testing fails, then you can modify the Remediation Plan to account for issues that were found and repeat the image creation and testing phases. This plan-test-revise cycle is repeated until a safe and viable replacement for the original system is produced. The validated Remediation Image is stored in the CRV can then be used to create a clean image which can then be deployed as the replacement production system.

Image Creation Overview

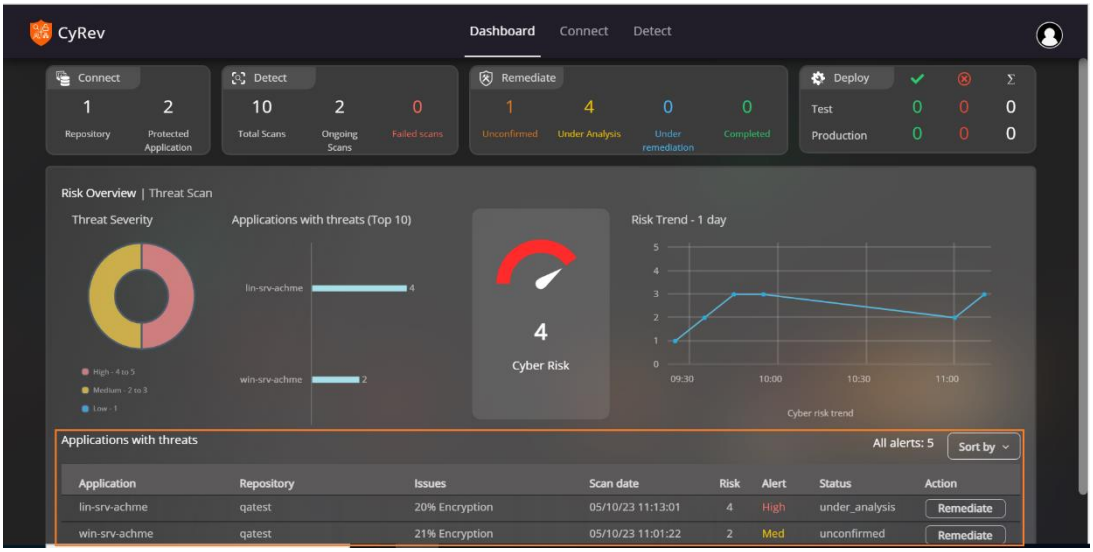
The process of creating a Remediation image is done in several steps:

1. An image you choose is recovered from the GCBDR image provider and attached to the Remediation host.
2. Using information from the CyRev UI, and from examining the attacked image, you generate the Remediation Plan, indicating dispositions for the files you wish to quarantine/delete, etc.
3. Using the Remediation Plan CyRev performs the operations in prescribes on the files in the attacked image and generates the Remediation Image, storing a copy in the CRV that can be used to generate VMs for testing and/or deployment as a replacement production system.

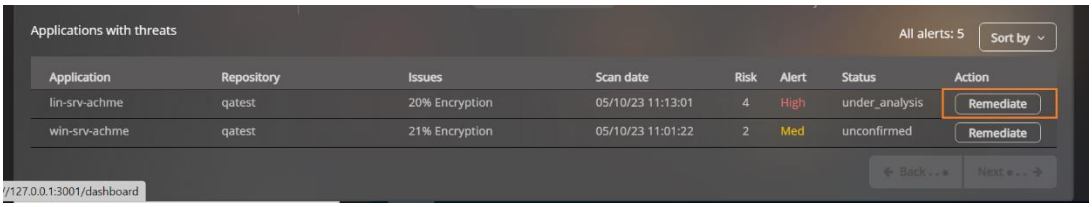
Creating a Remediation Plan and a Remediation Image

The Remediation Plan is action to remove ransomware from infected image and create clean image. You can see the list of all the files in the plan that can be remediate, quarantined or mark as safe. The remediated files are deleted from the generated Remediation Image. The quarantine files are excluded from the generated Remediation Image, and instead they are placed in the CyRev Quarantine Bucket for forensic purposes, or perhaps for further recovery actions. To create a Remediation Plan:

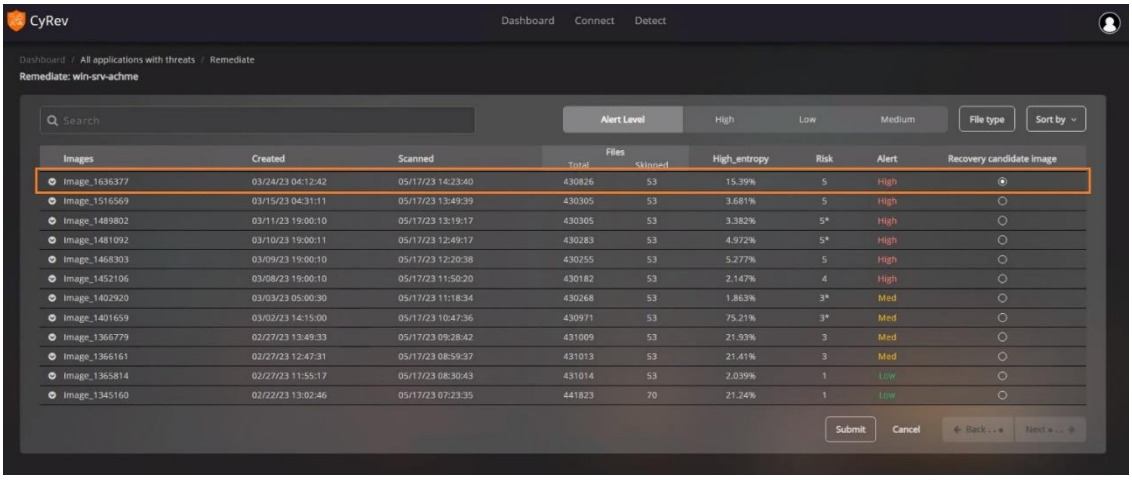
1. Connect to the CyRev Server Host via RDP using its cloud IP address.
2. On the Dashboard navigate to **Application with threat** table.



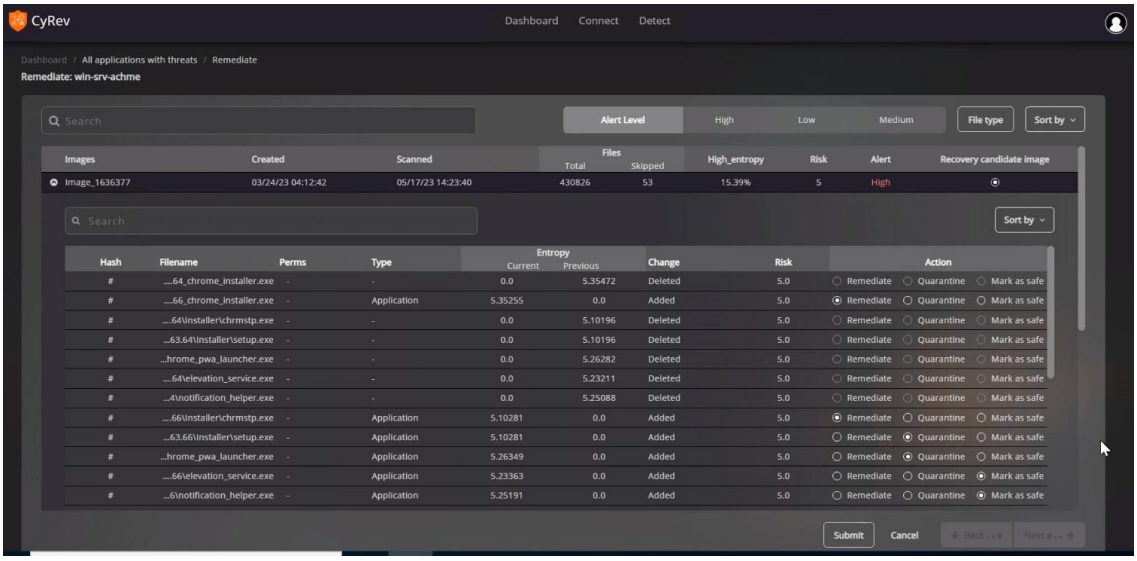
3. Select the application that you want to remediate and click **Remediate**.



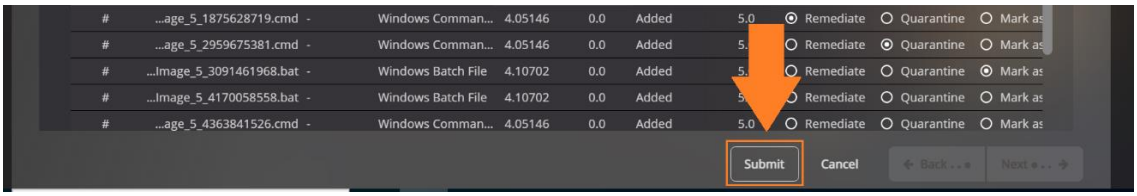
4. Check the radio button to select **Recovery candidate image**.



5. Select image row you have selected as recovery candidate image to see the files details. Based on your analysis mark the files as remediate, quarantine, or mark as safe. Use search bar to narrow down your search and select necessary files for the remediation plan and also helps you to search files across the images to select best recovery candidate image for your remediation plan.

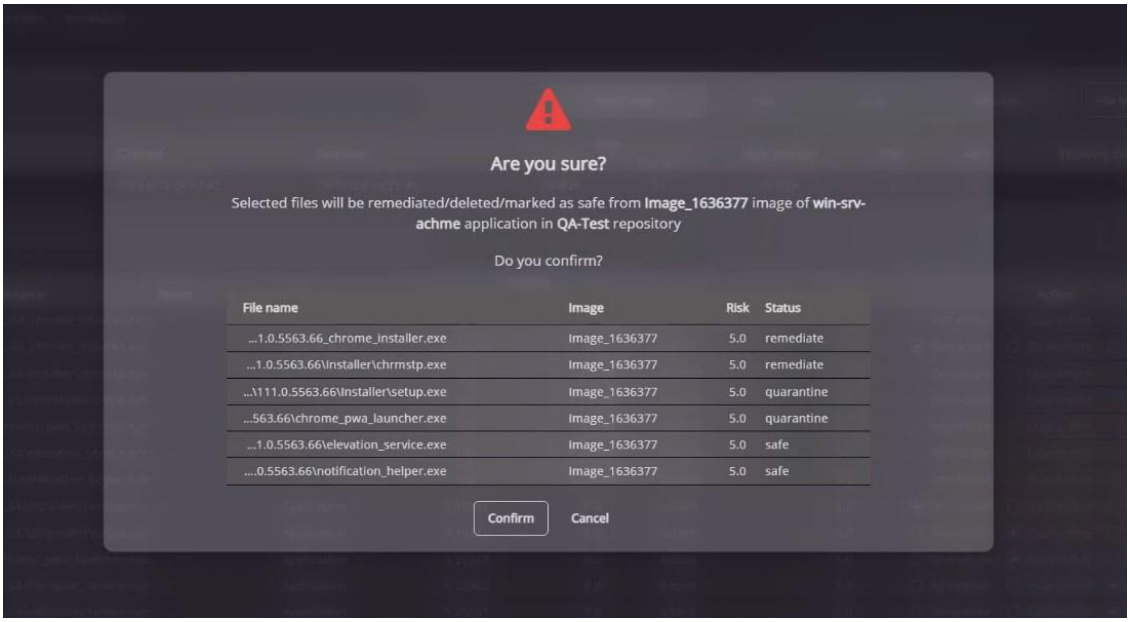


6. Select **Submit** button to complete remediation plan.



7. You will see the list of files that you have selected in remediation plan. (You can Cancel to revisit your remediation plan make changes accordingly).

8. Select **Confirm** to start remediation for application.



9. Open the command prompt and go to directory “C:\cyrev\bin” using following command:

```
cd c:\cyrev\bin
```

10. Execute the following command to start the remediation scheduler:

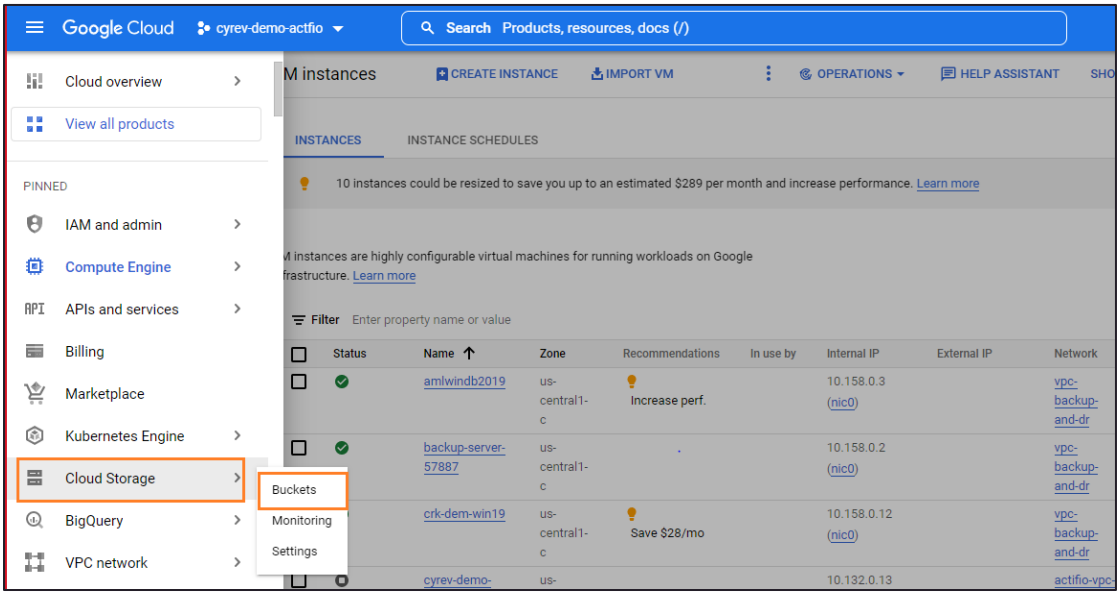
```
remediation_scheduler.exe
```

After submission of the application for remediation the status of the application changes to under remediation.

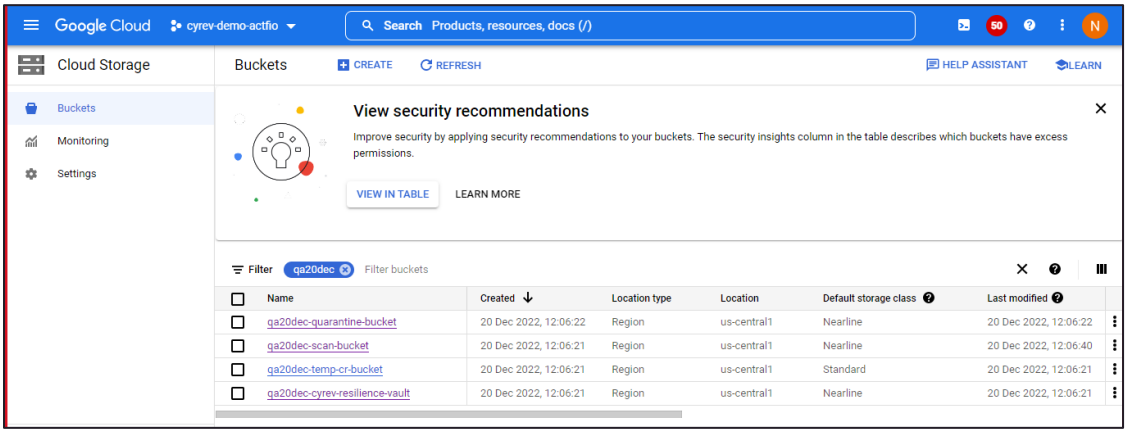
Upon the successful completion the clean image(s) will appear in the to the CR Vault bucket at the results for each application scan are stored in the CyRev Scan Bucket and Database.

following path:

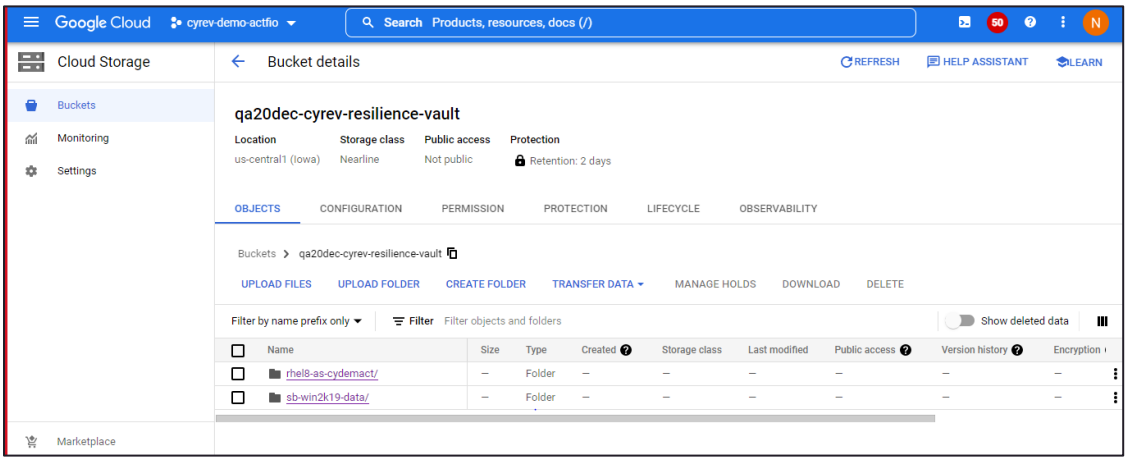
1. Login to your GCP Cloud Console.
2. Navigate to **Cloud Storage** → **Buckets**.



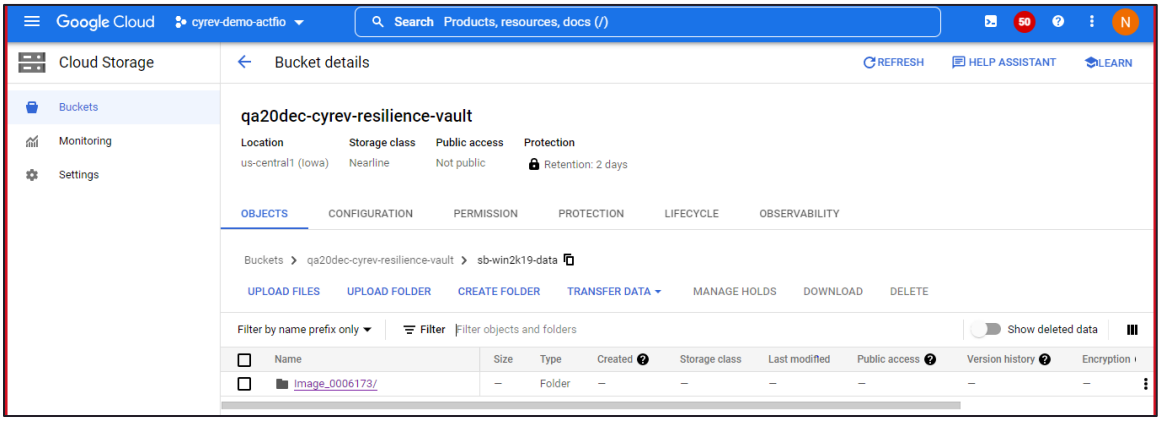
3. Select the **Cyber Resilience Vault** from the list.



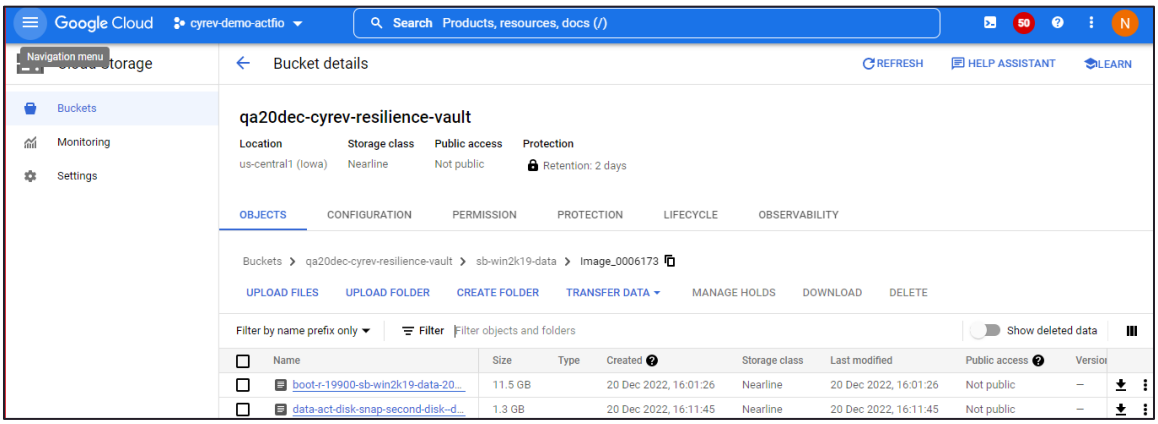
4. Select the **Application Name** of the system of which the image was created.



5. Select the **image name folder** of the specific system image used to create the Remediation Image.



6. You can see the **image name** that can now be used for testing and deploying cleaned VMs.



NOTE: Current retention policy period is 2 days for images stored in CRV. You can always review and update your retention policy as per Google Retention policies.

Files that were specified for quarantine are moved to the Quarantine Bucket, under the following path:

`<application-name>/quarantines/<timestamp>/<filepath>`

where:

- <*application-name*> is the name of the application, as specified in the configuration file
- <*timestamp*> is a string indicating when the command to create the Remediation Image was run; and
- <*filepath*> is the path to the quarantined file in its original image

For example, for the first quarantined file mentioned in the example above the file could be reached using the following URL:
`https://storage.cloud.google.com/qbucket/myapplication/quarantines/20220630T0507524243Z/corpdata/payroll.xls`

Testing A Candidate Image

Once clean images are saved in the CR Vault, they are ready to deploy. The most common next step is to test the cleaned image to be sure it's fully functional and contains unadulterated data. You can initiate a test and select the images to be run in the air-gapped testing environment.

To facilitate testing CyRev can deploy an Application Test VM based on a candidate Remediation Image you have created (see the section on Generating A Remediation Image.) Each Application Test VM is deployed into its own "air-gapped" area of the CyRev GCP project where it can be tested without worrying about re-infection from the outside, and where a possibly infected system can be run without propagating the infection across the project.

Prerequisites

Before creating a VM based on the candidate clean image for testing, you need to deploy a Test Management host which will coordinate the process of deploying an Application Test VM. Deployment of a Test Management Host is covered in the section on *Deploying A Testing Environment* in the *CyRev Getting Started Guide*. This creates an isolated VPC and the CyRev Test Management Host which is used to perform recovery and test operations.

The operations to create an Application Test VM is performed by a series of commands you will execute on the Test Management Host (see the section on Cyrev Commands.) These commands are controlled with configuration files. To fill out the configuration file you will need information about the CyRev deployment and the GCP environment in general. The necessary information is summarized with each command and can be filled in using the information described in the section on Command Configuration Files.

Application Test VM Deployment Parameters

The commands you will execute to perform the various steps for creating an Application Test VM use configuration files to specify parameters that control the operations. Using the steps below you need to fill out the following information in the configuration files before executing any CyRev commands:

1. Connect to the CyRev Server Host from your desktop via Remote Desktop Protocol (RDP) using its cloud IP address.
2. Connect to the CyRev Test Management Host via RDP using its cloud IP address from CyRev Server Host.
3. Open the configuration file you wish to update in a text editor; the file "`c:\cyrev\bin\Create-Test-Instance.json`" is provided as a default/template which you can copy or use directly. (See the section on CyRev

Command Configuration Files for details on using and managing configuration files.)

4. Change the below given values as per your project details and save the file:

Argument	Example	Description
Application Test VM Parameters		
"DeploymentName"	"uniquename"	Enter the unique name for test application VM that is to be created in GCP. **Note**: The length of the Deployment Name should not exceed 7-10 characters, and all characters must be lower case.
"ApplicationName"	"remedyapp-dev-dr"	The Application Name of the Remediation Image to use as the base of the deployed VM. (This is used as part of the path to the image in the CRV.)
"ImageName"	"Image_1483335"	The name of the Remediation Image you wish to use as the for the Application Test VM. You can obtain this Name from CR vault bucket.
"TargetRegion"	"asia"	The GCP region where the Application Test VM will be deployed.
"TargetZone"	"asia-south1-a"	The GCP zone where the Application Test VM will be deployed.
"Operation"	"create_test"	This parameter controls the creation and deletion of VM. Enter `create_test` to create application test VM. Enter `destroy_test` to delete the created application test VM.
"ImageType"	"windows"	Enter the image type as follows: `windows`: enter for windows image. `other`: enter for all other image types (For example: linux.).
CyRev Deployment Parameters		

Argument	Example	Description
"CRVaultBucket"	"s-crvault-bucket"	Enter the Name of the CR Vault Bucket name that created as a part of infrastructure deployment. You can obtain the name as instructed in Obtain CyRev Server Parameters section.
"RepoName"	"TestRepo1Name"	Provide the GCBDR repository name registered on Connect page of UI Dashboard.
"ServerName"	"dev-cyrev-server-host"	Enter CyRev Server Host name. You can obtain the name as instructed in Obtain CyRev Server Parameters section.
"ServerZone"	"us-central1-c"	Enter CyRev Server Host zone. You can obtain the zone as instructed in Obtain CyRev Server Parameters section.

5. For Application Test VM deployment, you will need to edit a second configuration file. Navigate to `C:\cyrev\bin\infra\services` on the Test Management Host.
6. Open the `variables.json` file in editor.
7. Change the below given values as per your project details and **save** the file.

Argument	Example	Description
customer_name	"uniquename"	Enter the same 'Deployment Name' provided in Create-Test-Instance.json.
project_id	"cyrev-poc-project"	The GCP project ID where the Application Test VM's environment will be provisioned. This must be the same project in which the Test Management Host was deployed.
subnet_test	"10.162.80.0/20"	Enter the unique & specific IP range in CIDR notation which will be used as the Application Test VM's network. Contact your project network administrator to get an available IP range as per your requirements.

Once these configuration files have been filled out have been installed you can deploy the Application Test VM.

Deploy an Application Test VM

The following steps use the parameters you configured in the prior section; be sure you have filled them all out before proceeding.

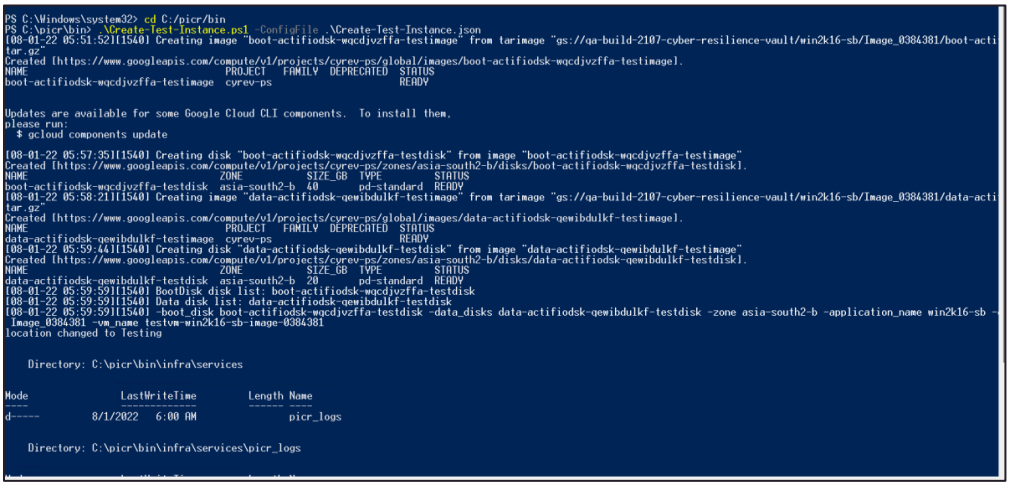
1. Connect to the CyRev Server Host from your desktop via RDP using its cloud IP address.
2. Connect to the CyRev Test Management Host via RDP using its cloud IP address from CyRev Server Host.
3. Open PowerShell 7 with administrative privileges to execute below steps.
4. Navigate to the directory `c:\cyrev\bin\` with the following command:

   ```
   cd c:\cyrev\bin\
   ```

5. Execute the following command, substituting the name of your configuration file:

   ```
   .\Create-Test-Instance.ps1 -ConfigFile .\Create-Test-Instance.json
   ```

   Example:



The Application Test VM is deployed in totally isolated environment, with no firewall rules, routes, etc. in place to give access to the newly deployed system to give it access to the outside world. You will need to create access as is appropriate for your application, your testing plans, your security requirements, etc. The deployed Application test VM has no external IP to access.

The most basic access can be granted through a peer relationship between the deployed Application Test environment and VPC of your choice. For example, it might be convenient to connect to the Application Test VM from the Jump Server deployed within the same VPC.

Once deployment has completed and any necessary access has been granted you are free to test the system as you see fit. If you encounter problems with the contents of the

base image (ie, the Remediation Image you specified) you can modify the Remediation Plan that generated it, create a new Remediation Image and deploy a new Application Test VM to repeat testing.

Once you have a clean and valid Remediation Image that passes your tests, you can deploy it in its own safe environment by following the instructions in the next section.

Deploying A Production Image

Once images are tested, they are ready to deploy. You can initiate a deployment for production images and select the images to be run in the air-gapped production environment.

To facilitate production image deployment CyRev can deploy an Application Production VM based on a tested Image you have tested (see the section on Testing A Candidate Image.) Each Application Production VM is deployed into its own “air-gapped” area of the CyRev GCP project where it can be deployed without worrying about re-infection from the outside, and where a possibly infected system can be run without propagating the infection across the project.

Prerequisites

Before creating a VM based on the tested clean image for production, you need to deploy a Production Deployment Management host which will coordinate the process of deploying an Application Production VM. Deployment of a Production Deployment Management Host is covered in the section on *Deploying A Production Deployment Management Host* in the *CyRev Getting Started Guide*. This creates an isolated VPC and the CyRev Production Deployment Management Host which is used to perform production operations.

The operations to create an Application Production VM is performed by a series of commands you will execute on the Production Deployment Management Host (see the section on Cyrev Commands.) These commands are controlled with configuration files. To fill out the configuration file you will need information about the CyRev deployment and the GCP environment in general. The necessary information is summarized with each command and can be filled in using the information described in the section on Command Configuration Files.

Application Production VM Deployment Parameters

The commands you will execute to perform the various steps for creating an Application Production VM use configuration files to specify parameters that control the operations. Using the steps below you need to fill out the following information in the configuration files before executing any CyRev commands:

1. Connect to the CyRev Server Host from your desktop via RDP using its cloud IP address.
2. From the CyRev Server Host connect to the CyRev Production Deployment Management Host via RDP using its cloud IP address.
3. Open the configuration file you wish to update in a text editor; the file "`c:\cyrev\bin\Create-Production-Instance.json`" is provided as a

default/template which you can copy or use directly. (See the section on CyRev Command Configuration Files for details on using and managing configuration files.)

4. Change the below given values as per your project details and save the file:

Argument	Example	Description
Application Production VM Parameters		
"DeploymentName"	"uniquename"	Enter the unique name for production application VM that is created in GCP. Note: The length of the Deployment Name should not exceed 7-10 characters, and all characters must be lower case.
"ApplicationName"	"remedyapp-dev-dr"	The Application Name of the Tested Image to use as the base of the deployed VM. (This is used as part of the path to the image in the CRV).
"ImageName"	"actifiosky.persistent.co.in_Image_1483335"	The Image name you wish to use as the image for the Application Production VM.
"TargetRegion"	"asia"	The GCP region where the Application Production VM will be deployed.
"TargetZone"	"asia-south1-a"	The GCP zone where the Application Production VM will be deployed.
"Operation"	"create_production"	This parameter controls the creation and deletion of VM. Enter `create_production` to create application production VM. Enter `destroy_production` to delete the created application production VM.
"ImageType"	"windows"	Enter the image type as follows: `windows`: enter for windows image. `other`: enter for all other image types (For example: linux).

Argument	Example	Description
CyRev Deployment Parameters		
"CRVaultBucket"	"s-crvault-bucket"	Enter the Name of the CR Vault Bucket name that created as a part of infrastructure deployment. You can obtain the name as instructed in Obtain CyRev Server Parameters section.
"RepoName"	"TestRepo1Name"	Provide the GCBDR repository name registered on Connect page of UI Dashboard.
"ServerName"	"dev-cyrev-server-host"	Enter CyRev Server Host name. You can obtain the name as instructed in Obtain CyRev Server Parameters section.
"ServerZone"	"us-central1-c"	Enter CyRev Server Host zone. You can obtain the zone as instructed Obtain CyRev Server Parameters section.

5. For Application Production VM deployment, you will need to edit a second configuration file. Navigate to `C:\cyrev\bin\infra\services` on the Production Deployment Management Host.
6. Open the `variables.json` file in editor.
7. Change the below given values as per your project details and **save** the file.

Argument	Example	Description
customer_name	"uniquename"	Enter the same 'Deployment Name' provided in Create-Production-Instance.json.
project_id	"cyrev-poc-project"	The GCP project ID where the Application Production VM's environment will be provisioned.
subnet_production	"10.162.80.0/20"	Enter the unique & specific IP range in CIDR notation which will be used as the Application Production VM's network. Contact your project network administrator to get an available IP range as per your requirements.

Once these configuration files have been filled out, you can deploy the Application Production VM.

Deploy an Application Production VM

The following steps use the parameters you configured in the prior section; be sure you have filled them all out before proceeding.

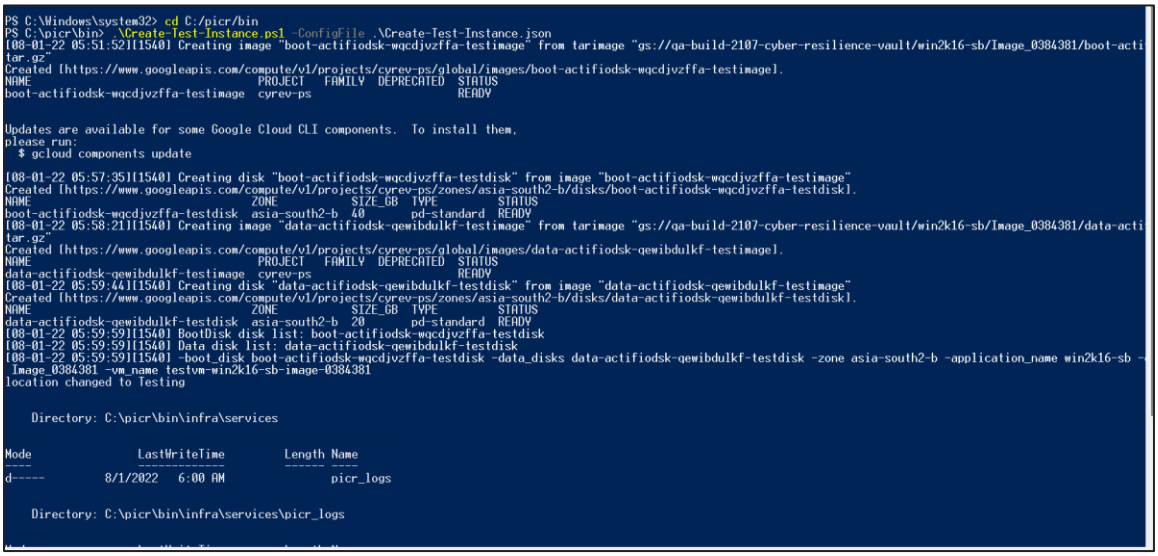
1. Connect to the CyRev Server Host from your desktop via RDP using its cloud IP address.
2. Connect to the CyRev Production Deployment Management Host via RDP using its cloud IP address from CyRev Server Host.
3. Use PowerShell 7 with administrative privileges to execute below steps.
4. Navigate to the directory `c:\cyrev\bin\` with the following command:

   ```
   cd c:\cyrev\bin\
   ```

5. Execute the following command, substituting the name of your configuration file:

   ```
   .\Create-Production-Instance.ps1 -ConfigFile .\Create-Production-Instance.json
   ```

   Example:



The Application Production VM is deployed in totally isolated environment, with no firewall rules, routes, etc. in place to give access to the newly deployed system to give it access to the outside world. You will need to create access as is appropriate for your application, your testing plans, your security requirements, etc. The deployed Application Production VM has no external IP to access.

The most basic access can be granted through a peer relationship between the deployed Application production environment and VPC of your choice. For example, it might be

convenient to connect to the Application Production VM from the Jump Server deployed within same VPC.

The Application Production VM is deployed, and you are free to attach it your organization infrastructure to make it up and running.

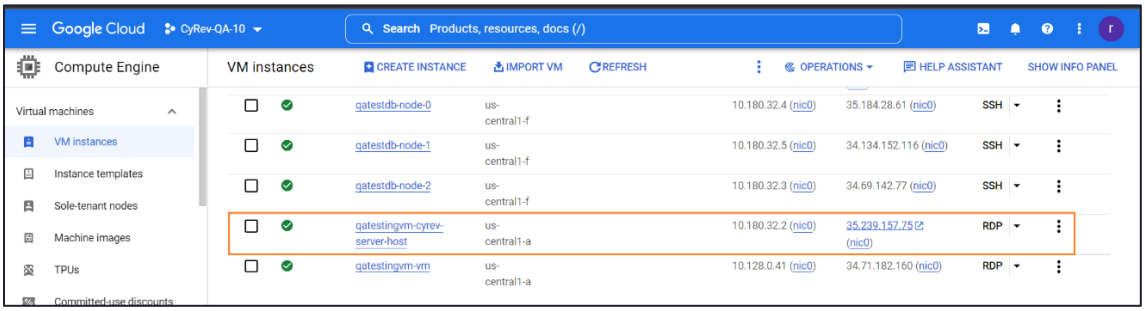
Reference

This Section contains the additional information about the information you may need to refer during the process.

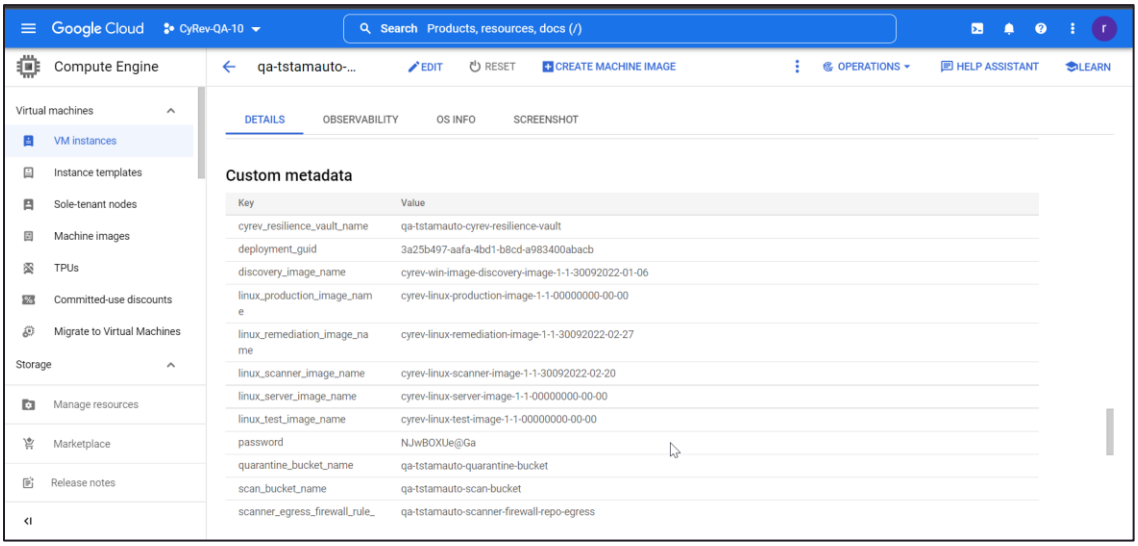
Obtain CyRev Server Parameters

To use the CyRev dashboard you need to obtain the username and password for your UI. Follow the steps to obtain your UI credentials:

1. Navigate to **Compute Engine** → **VM instances** on your GCP cloud console.
2. Select the VM with deployment name with **cyrev-server-host** from the deployed VM's.



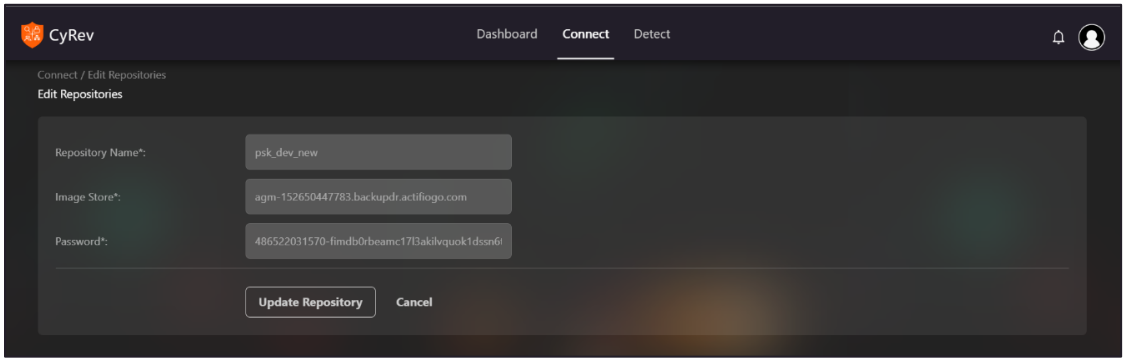
3. Scroll-down to the **Custom Metadata** section and make note of the parameters you want.



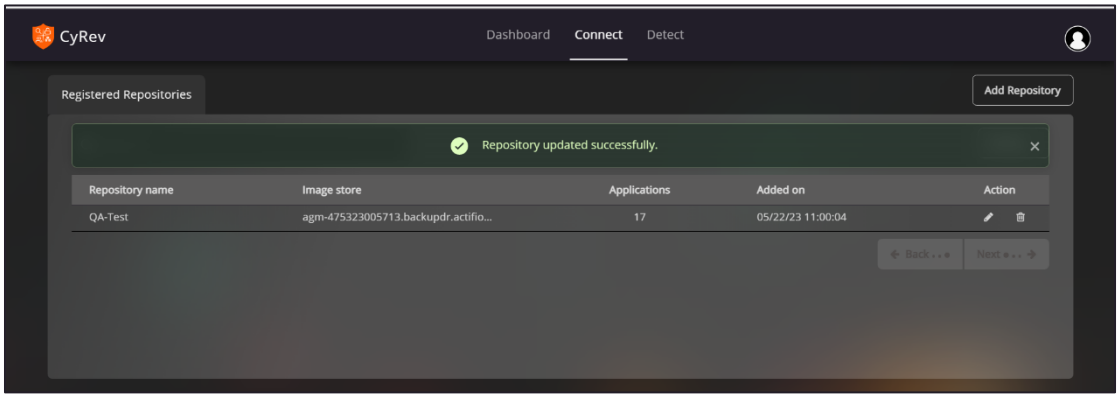
Edit Image Provider Repository Details

You can always update the Image Provider parameters.

1. Log in to CyRev dashboard.
2. Click **Connect**.
3. Select the repository for which you want to update and select the  icon in action column. Edit repository page opens.
4. Update the **Image store** address and **Password**.



5. Select **Update Repository** to update information. You will be notified after the successful update.



Contacting Support

Persistent Systems Software support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by being able to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Submit enhancement requests online
- Download software patches
- Look up Persistent support contacts
- Enter into discussions with other software customers
- Research and register for software training

To access the Self-serve knowledge base, visit the Persistent System Support home page at

https://support.persistent.com/hc/en-us

Most of the support areas require that you register on the Persistent Systems Support Portal. Many also require a support contract.

To register an account at the Persistent Systems, Support Portal, visit

https://support.persistent.com/hc/en-us

To know more about registration process at Persistent Systems support portal, visit

https://support.persistent.com/hc/en-us/articles/202042570-New-user-registration-process

About Persistent

With over 13,500 employees around the world, Persistent Systems (BSE & NSE: PERSISTENT) is a global services and solutions company delivering Digital Engineering and Enterprise Modernization.

www.persistent.com

India

Persistent Systems Limited

Bhageerath, 402,

Senapati Bapat Road

Pune 411016.

Tel: +91 (20) 6703 0000

Fax: +91 (20) 6703 0008

USA

Persistent Systems, Inc.

2055 Laurelwood Road, Suite 210

Santa Clara, CA 95054

Tel: +1 (408) 216 7010

Fax: +1 (408) 451 9177

Email: info@persistent.com