ACSIA Administration Guide

This guide will walk you through the basics of using ACSIA, how to install it, administrate and interactively monitor your security.

What is ACSIA?

ACSIA is an **A**utomated **C**yber **S**ecurity **I**nteractive **A**pplication which enables organizations to protect themselves against malicious attacks and unauthorized entities.

Network Requirements

Like any other software application, ACSIA needs some preliminary requirements to be satisfied in order to communicate with connected clients and to perform its tasks. ACSIA is an on-premise application server and therefore we strictly recommend customers to **restrict the access to the server** at network level only to relevant people who are designated to administer and manage security. Restricting and securing the ACSIA server at network level is beyond ACSIA's control as these settings are managed by customer's own on site Firewalls. Please create a white-list on your firewall and allow only those people who need access to ACSIA to administer the server. It is purely the customers responsibility to secure ACSIA at their network perimeter.

Web UI Network ports

In order to access ACSIA's Web UI from your device/PC the following ports need to be opened in inbound on your firewalls:

- HTTP: 8080 (redirects to HTTPS:8443)
- HTTPS: 8443, 5601.

ACSIA client server ports

ACSIA operates via the following communications ports with its clients and therefore the following specific rules will need to be created on your firewalls:

- TCP->Inbound: 5044
- TCP->Outbound: 22

Furthermore, ACSIA will need internet access. This access is only for outbound and during the installation process. After completion of the installation this can be restricted if required, despite being considered harmless outbound access.

The only outbound connectivity **required** by ACSIA after install is TLS-enabled connectivity to port 5150 on license.acsia.io. This connectivity is required at all times, although ACSIA allows up to 48 hours of connection loss before it enters unlicensed state.

Also make sure that your ACSIA server comes with **mail server** such as `postfix` installed and enabled (usually the system comes automatically with this requirement but it is worth checking in case of any issue with email notifications). This is an imperative requirement in order for ACSIA to send **email notifications**, and to send the initial password reset link to the first ACSIA user.

Once the above network requirements are satisfied for ACSIA's server IP address, the installation process can begin as follows.

Email Notification Requirements

In order for ACSIA to be able to send email notifications the instance itself will need the ability to communicate with your mail server and the emails originating from ACSIA to be white-listed on your spam filters (if you have any).

For those who will be using non-business emails, i.e. Gmail, Yahoo, Microsoft Outlook etc. we recommend to check your spam filters.

We also advise creating a DNS record under your domain for the ACSIA server to help to white list the emails to be filtered by the aforementioned private email providers.

Use an email account to receive notifications

It is possible to use an email account (ie. Gmail, Yahoo, etc.) to receive notifications via mail.
To activate it:

1. Create a file named `custom.properties`
2. Put the following content (with properties set correctly):

```
# SMTP server host. For instance, `smtp.example.com`
spring.mail.host=

# SMTP server port (same port for both properties
spring.mail.port=
spring.mail.properties.mail.smtp.socketFactory.port=

# Login user of the SMTP server
spring.mail.username=

# Login password of the SMTP server
spring.mail.password=

spring.mail.properties.mail.smtp.auth=true
spring.mail.properties.mail.smtp.socketFactory.class=javax.net.ssl.SSLSocketFactory
spring.mail.properties.mail.smtp.socketFactory.fallback=false;
```

3. set/update properties (see below)

How to set/update property script

- **Installation time**

You can set the properties file at **installation time** via `-p` argument:

```
./acsia_install -p custom.properties
```

- **Already installed instance**

You can also choose to override it by adding the file on **already installed** ACSIA instances. **A restart of the engine service is required** (`acsia_restart` script can be used)

```
cp custom.properties "$ACSIA_ENG/resources/custom.properties"
```

Reset notification mail service

If you have set an account mail to receive ACSIA notifications and you want to switch back to the default you have to create a file named `custom.properties` and populate with the following properties:

```
spring.mail.protocol=smtp
spring.mail.host=localhost
spring.mail.port=25
spring.mail.properties.mail.smtp.auth=false
spring.mail.properties.mail.smtp.starttls.enable=false
spring.mail.properties.mail.smtp.connectiontimeout=5000
spring.mail.properties.mail.smtp.timeout=5000
spring.mail.properties.mail.smtp.writetimeout=5000
```

Then place the file in the relevant directory and restart the engine:

```
cp custom.properties "$ACSIA_ENG/resources/custom.properties" && acsia_restart
```

ACSIA Server installation

The environment ACSIA in which is installed should reflect the following minimum specs:

- CentOS or RHEL (7) Linux
- A user account called "acsia" with sudo passwordless privileges
- 8GB RAM 2-4vCPU (for ~10 to 30 clients so scale up accordingly)
- Storage, depending on your needs and the amount of logs to store

Please also ensure that user-level open file limits are set to sufficiently high values - we recommend 65535+ soft/hard. These must be permanent user-level limits, set in `/etc/security/limits.conf` with a subsequent **system reboot** to take effect.

System patching

Automated system updates are **not supported** by ACSIA, so if `yum-cron` is enabled on the server, it will be disabled. Please ensure the system is manually kept patched. When performing system patching, please ensure you take ACSIA offline (`acsia_stack_stop`), perform any system patching, then bring ACSIA back online (`acsia_stack_start`).

Kernel monitoring requirements

NB: Please ensure that the server is fully patched, and that `kernel-devel-$(uname -r)` package is available. This is required for kernel-level monitoring to be installed. This can be checked by e.g. the below:

```
$ yum list kernel-devel-$(uname -r) >/dev/null 2>&1 && echo available || echo not
available
not available
```

This is required for *all* servers for kernel monitoring. Note that on some cloud provider servers, this requires a full update and subsequent reboot e.g. `yum update; reboot`. Servers should be prepared *before* ACSIA is installed or its shippers deployed to clients.

Deployment

Customers with regular ACSIA licenses will receive instructions to download the main installation script and an additional file containing specific steps to be followed.

The files are:

```
- acsia_download
- README.md
```

The `README.md` file contains the initial setup installation instructions. For ACSIA server side installation there is a single main instruction to follow.

Enterprise Version

Steps to perform in bash `acsia` account

```
    ./acsia_download && ./acsia_install
```

Lite Version

Steps to perform in bash `acsia` account

```
    ./acsia_download  --repository lite-releases --user username
    ./acsia_install
```

And that is it - if all preliminary requirements are satisfied as per the guidelines, you will have ACSIA server installed and started within ~10 minutes. In case of any issue please refer to the Troubleshooting section below.

NB: do not attempt to access the ACSIA UI before the initial registration email has been sent, as ACSIA is still initialising.

Download prompts

To download ACSIA, you will be prompted for a username and password - these will be provided with your licensing information.

Installation prompts

During the installation, you will be prompted to enter some information to configure your install. These are as follows:

- Initial ACSIA **user email address**: this is the email address of the initial user added to ACSIA. This can also be set from the command-line with option `-u` or `--username` (check `./acsia_install --help`)
- **IP address**: If you are using an internal IP address for inter-server communication (rather than an external), please enter this here. Otherwise, ACSIA will determine the server's public IP address and use this. This can also be set from the command-line (i.e. `-i10.20.3.254` or `--ip-address 10.20.3.254`)
- **SSH 2-Factor**: See the notes on 2-Factor Authentication below. This can also be set from the command-line `(-s or --ssh-2fa; true|false)`
- **ACSIA 2-Factor**: See the notes on 2-Factor Authentication below. This can also be set from the command-line `(-e or --engine-2fa; true|false)`

NOTE: Please be aware that if you change the IP address of ACSIA manually from the configuration file you will break the all communications systems with the clients connected (if any added). Also trying to change/amend manually the server IP address on individual clients connected to ACSIA will not help to restore the communication between the two since ACSIA generates SSL cert based private and public keypair between the server and it's clients. Therefore, if by accident you change the IP address of ACSIA server the easiest fix would be to run 'acsia_update_ip' from ACSIA server itself.

At the end of the process you will see a summary of how to access the ACSIA Web UI including account access details.
If you have enabled `SSH 2-Factor Authentication`, you will also be presented with an ASCII QR code, compatible with e.g. the Google Authenticator mobile application.
The initial user will also receive an email containing their password reset link (to set their initial password), and a 2FA QR code if applicable (pls check your spam if you don't see any email coming from ACSIA).

You will find all of ACSIA's automated services and troubleshooting scripts under `bin` directory in `$ACSIA_HOME`

Deployment of custom SSL certificates

By default, ACSIA generates self-signed SSL certificates for HTTPS browsing. These are configured for the IP address of the instance (external or local if provided). If you wish to deploy ACSIA with your own SSL certificates (e.g. those generated by certbot), this can be done at install-time or afterwards, since ACSIA v.2.1.1. In order to deploy SSL certificates, they need to be provided in two files - the private key, and the public certificate, both as .pem formatted files.

1. Install-time deployment: when installing using the `acsia_install` script, simply add the extra command-line arguments: `--certificate /path/to/cert.pem`, `--key /path/to/key.pem` and `--domain my.domain.com`. All three must be present, or the installer will exit with an error.
2. Post-install deployment: a new script `acsia_deploy_ssl_certs` has been provided, and accepts the same `--certificate`, `--key`, and `--domain` arguments as the install script. Once these have been deployed, you will need to execute `acsia_stack_restart` for all components to pick up these new certificates.

Amazon AWS Marketplace

For those who purchases ACSIA via Amazon AWS Marketplace, the installation process is straight forward.

Once you start the instance with ACSIA all you need to do is to login first via SSH as `acsia` user and run the following script by entering your details:

`./acsia_configure`

The next step would be to login on ACSIA UI and enter the license supplied as first step.

ACSIA Service Daemons

As mentioned above, all of ACSIA's executables and daemons to start/stop services and perform troubleshooting can be found in `bin` folder in `$ACSIA_HOME`.

To check if all of ACSIA services running:

```
acsia_stack_status
```

To start ACSIA services:

```
acsia_stack_start
```

To stop ACSIA services:

```
acsia_stack_stop
```

ACSIA is a modular product and therefore there are several service scripts in individual modules for performing better troubleshooting. These modular service scripts will rarely be needed so the above shown service scripts should suffice for most work on ACSIA.

NOTE: After installation of ACSIA, to be able to execute above mentioned commands under `$ACSIA_HOME/bin`, immediately after installation we recommend you either logout and login in order for your session to pickup environment variables or simply source them by running `source "$HOME/.bashrc"`.

Updating ACSIA

Enterprise Edition Update

Steps to perform in bash `acsia` account

```
acsia_update
```

Lite Edition Update

Steps to perform in bash `acsia` account

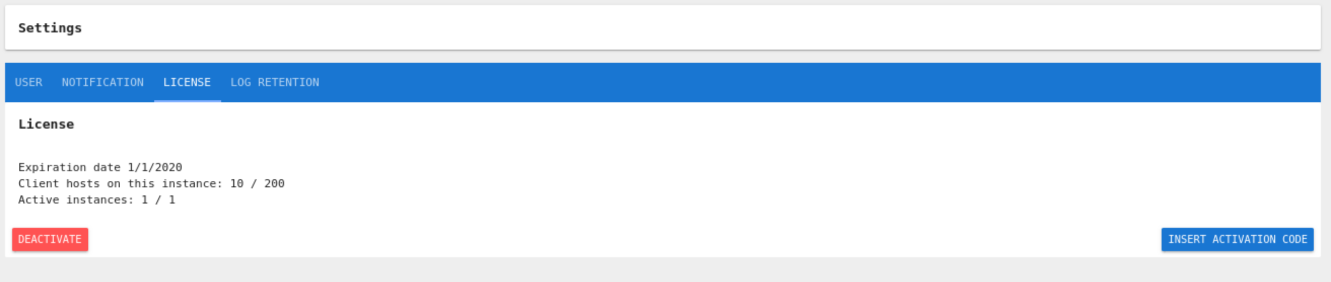
```
acsia_update --repo 'lite-releases' --user usaername
```

For both editions you will require your ACSIA username and passwords to get updates. Those parameters are usually supplied within the same email where you receive your license instructions.

License Activation

After having purchased ACSIA you will receive your license instructions and if you haven't received please contact us via https://acsia.io/support.html

ACSIA will allow you login on its web UI but you won't be able to do anything until you activate the license. To do so, all you need is to copy the `license code` and go to top right user menu by clicking on `Settings` and then select `License` where you can add the license.



Settings

License

Expiration date 1/1/2020
Client hosts on this instance: 10 / 200
Active instances: 1 / 1

Just activate the license and you are all set and good to go.

For those who purchases ACSIA from Amazon AWS Marketplace this section can be disregarded.

NOTE: The license can be requested via contact form at https://acsia.io in case you have not received your license (or by contacting support@acsia.io).

2-Factor Authentication

To better secure your ACSIA installation, two optional layers of 2-Factor Authentication (2FA) are available - SSH and ACSIA Web UI.

SSH

To better protect your ACSIA server, you may apply 2FA to the `acsia` user for SSH entry. This is configured during installation, or may be dis/enabled at any time using the `mfa_ssh_install` and `mfa_ssh_uninstall` scripts in the `$ACSIA_HOME/bin` directory. When enabled, the `acsia` system user will have a Time-based One-Time Password (TOTP) secret key generated. Any users attempting to initiate SSH sessions as this user will have to use this secret to generate One-Time Passwords (OTPs) - we recommend the Google Authenticator application, however, any application capable of generating TOTPs will work. A QR code is printed to screen when this is set up.

NOTE: when this is enabled, **only** key-based authentication can be used for session authentication. Password + 2FA is not supported, as this causes unintended side-effects.

After the installation is finished, or when executing `mfa_ssh_qrcode`, the QR code will be displayed to the terminal.

ACSIA Web UI

Although the SSH 2FA can be optionally enabled or disabled, we strictly recommend that the 2FA is to be enabled for UI. Therefore we recommend to apply per-user 2FA. If this is set, all users will be provided with a QR code (which should be scanned by e.g. Google Authenticator, FreeOTP, etc.), and must present the TOTP key each time they log in to the application.

Kibana dashboards authentication

To protect the Kibana dashboards (hosted on SSL-enabled port 5601) from external users, password authentication is enabled. Whenever a user attempts to view Dashboards, they must present their ACSIA username/password.

Preparing Your Servers to Get Ready for ACSIA

An example of creating ACSIA service user for RedHat/Debian systems:

NOTE: If your IT infrastructure is hosted on **Google Cloud** (Metadata Page) you can skip the following steps by just adding ACSIA ssh-key to your project from Google console. AWS has similar setup that can be done with auxiliary of OpsWorks.

From your server shell console as root or equivalent:

An example of creating ACSIA Service User on `CentOS` or `RHEL` systems

```
useradd -m -d /home/acsia -c "ACSIA Service User" -G wheel acsia
```

Followed by enabling **sudo passwordless permissions** by editing `/etc/sudoers` file and uncomment the following string:

```
%wheel ALL=(ALL:ALL) NOPASSWD:ALL
```

An example of creating ACSIA Service User on `Debian` or `Ubuntu` systems

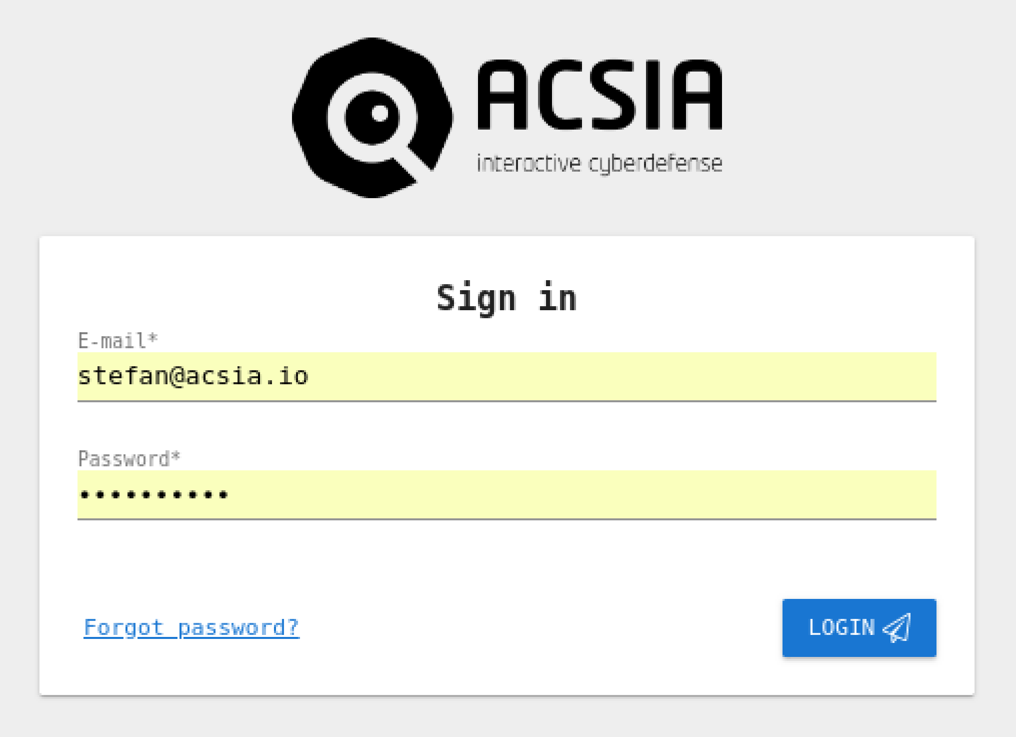
```
adduser acsia
```

```
usermod -aG sudo acsia
```

And than check in `/etc/sudoers` and make sure to uncomment following string (add if you don't have it):

```
%sudo ALL=(ALL:ALL) NOPASSWD:ALL
```

Once completed the service user creation on your servers the next step is to start using ACSIA Web UI as shown in the screen-shot:

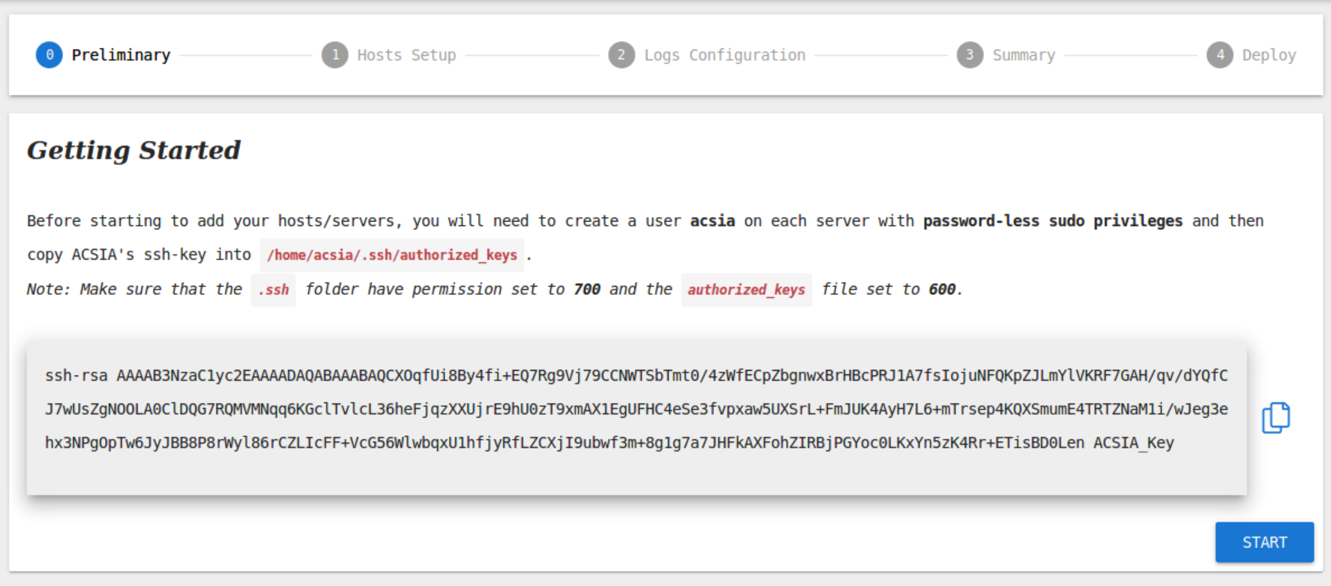


Sign in

E-mail*
stefan@acsia.io

Password*
••••••••••

ssh-key



Getting Started

Before starting to add your hosts/servers, you will need to create a user **acsia** on each server with **password-less sudo privileges** and then copy ACSIA's ssh-key into `/home/acsia/.ssh/authorized_keys` .

*Note: Make sure that the `.ssh` folder have permission set to **700** and the `authorized_keys` file set to **600**.*

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCXOqfUi8By4fi+EQ7Rg9Vj79CCNWTSbTmt0/4zWfECpZbgnwxBrHBcPRJ1A7fsIojuNFQKpZJLmYlVKRF7GAH/qv/dYQfC
J7wUsZgNOOLA0ClDQG7RQMVMNqq6KGclTvlcL36heFjqzXXUjrE9hU0zT9xmAX1EgUFHC4eSe3fvpxaw5UXSrL+FmJUK4AyH7L6+mTrsep4KQXSmumE4TRTZNaM1i/wJeg3e
hx3NPgOpTw6JyJBB8P8rWyl86rCZLIcFF+VcG56WlwbqxU1hfjyRfLZCXjI9ubwf3m+8g1g7a7JHFkAXFohZIRBjPGYoc0LKxYn5zK4Rr+ETisBD0Len ACSIA_Key
```

Copy/export the ssh-key as suggested in the brief instructions shown in the above screen-shot and import into the ACSIA Service User created (on each server) earlier paying attention to steps in relation to `.ssh` and `authorized_keys` file permissions.

```
echo "paste here the key" > .ssh/authorized_keys

chmod 700 .ssh && chmod 600 .ssh/authorized_keys
```

**NOTE: The ssh-key we refer to is a 4096 bit randomly generated by ACSIA during it's installation and therefore it can be changed by user, generated new one or replaced by any other keypairs (i.e. AWS EC2 keypair.pem etc). It is entirely at user discretion.

Windows Client implementation for ACSIA Server

ACSIA is currently running in beta mode on Windows systems and therefore several steps and requirements need to be performed manually as follows due to this environment being still under development.

To manage Windows servers, **WinRM** needs to be installed, running, and listening on the Windows servers. The instructions below are *examples* provided by Ansible; 4Securitas do not provide support for these. ACSIA currently only supports local administrative Windows accounts (no Active Directory accounts supported at the moment), and only currently supports password authentication - no Certificate/CredSSP/Kerberos etc.

Requirements

- PowerShell 3.0+
- Windows Server 2008 SP1+ (or Windows 7 SP1)
- Local `acsia` account with administrative privileges

Setup

Client setup

Ansible setup

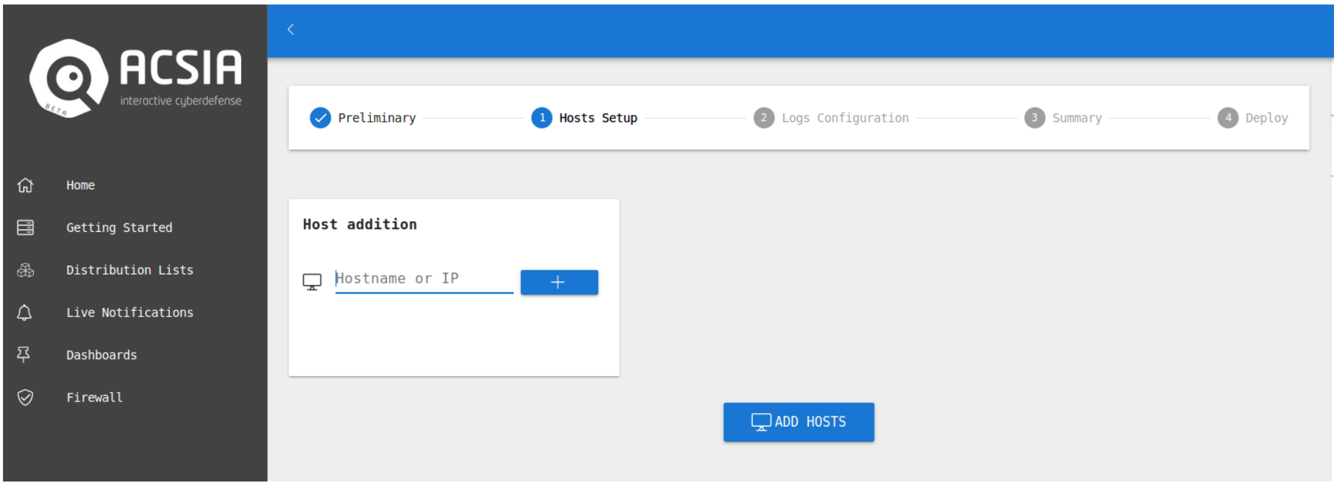
This script must be executed in order to allow Windows servers to use ansible via **winRM**

```
$url = "https://raw.githubusercontent.com/jborean93/ansible-windows/master/scripts/Upgrade-PowerShell.ps1"
$file = "$env:TEMP\Upgrade-PowerShell.ps1"
$username = "Administrator" # use a user account with administrative permissions
$password = "Password" # and their password
(New-Object -TypeName System.Net.WebClient).DownloadFile($url, $file)
Set-ExecutionPolicy -ExecutionPolicy Unrestricted -Force
# version can be 3.0, 4.0 or 5.1
&$file -Version 5.1 -Username $username -Password $password -Verbose
$url = "https://raw.githubusercontent.com/jborean93/ansible-windows/master/scripts/Install-WMF3Hotfix.ps1"
$file = "$env:TEMP\Install-WMF3Hotfix.ps1"
(New-Object -TypeName System.Net.WebClient).DownloadFile($url, $file)
powershell.exe -ExecutionPolicy ByPass -File $file -Verbose
$url = "https://raw.githubusercontent.com/ansible/ansible/devel/examples/scripts/ConfigureRemotingForAnsible.ps1"
$file = "$env:TEMP\ConfigureRemotingForAnsible.ps1"
(New-Object -TypeName System.Net.WebClient).DownloadFile($url, $file)
powershell.exe -ExecutionPolicy ByPass -File $file
```

NOTE: Please be aware that some of the windows clients may require rebooting depending on their update and patch level.

Adding Hosts via ACSIA Web UI

Login into ACSIA Web UI and click on `Getting Started` from the menu on the left side bar. Just click on `START` (just below ACSIA's ssh-key) and you will be forwarded to next window where you may add hosts on an IP or Hostname basis. It is best practice is to **use IP addresses** to add your server as the remaining details of hosts including hostname will be automatically retrieved and placed by ACSIA. So just add the IP of the Hosts/Servers and proceed. When you finish adding all, push "ADD HOSTS"



After completing `Preliminaries` and `Hosts Setup` the next step is `Logs Configuration` screen, where you can add your custom web server or web application logs (always use the absolute path of the logs). The next step is the validation of the logs entered, if any are non-existent this will be highlighted so that you can correct or remove if it was a mistake or a typo and so on. **NB**: the paths must be to the logfiles themselves; **symlinks to files are unsupported**.

Once completed the logs configuration there will be a `Summary` shown in the next screen to make sure that you have everything entered followed by `Deploy` where you actually start to deploy ACSIA and connect to your servers. This will take a few minutes and you will be eventually shown a dialog message that reports the completion of the enrollment procedure.

One completed, your servers are set to be constantly monitored in real time for security issues, threats and anomalies.

NOTE: If you are running **Ubuntu Xenial** the servers that you intend to connect to ACSIA, we recommend that you install python2.7 as Ubuntu Xenial ships with python3 only and it doesn't have previous versions installed. So please make sure that you have python 2 installed and available at `/usr/bin/python`.

Container-specific details

ACSIA is container-aware, and will automatically track kernel events etc. within containers. However, if you are running application/webservers within containers, and wish for these logs to be monitored, they must be made available to the host. As normal, ACSIA does not support symlinks to log files - it must be the full path to the file.

Troubleshooting and Common Issues

Issue: Some Ubuntu versions don't have python2

`yum install python2`

Issue: Registered user not receiving email with the account details

If you are experiencing any issues in receiving the email containing account details and login it is likely the case that your email provider bouncing the email, placing into spam or quarantine area or your network infrastructure is not allowing the server to dispatch emails. ACSIA avail of postfix as MTA and therefore please check your infrastructure requirements and accordingly setup the necessary configuration.

Issue: Deployment Failed

While trying to connect your server to ACSIA, if you are experiencing an issue with the message `Deployment Failed` probably one of ACSIA's modules are failing to install on your client server side. One of the very common issues could be that your server is not able to reach out the kernel layer module and therefore the module fails.

To solve this issue login into your client server, become `acsia` user and run:

```
curl https://s3.amazonaws.com/download.draios.com/stable/install-falco | sudo bash
```

If the installation completes then re-deploy the client on ACSIA UI.

Otherwise, if no output given or an output containing message `W: Failed to fetch` it means you are experiencing network issues to reach out to server where the module is located. Please check your outbound connection i.e. firewall rules and make sure the server is allowed reach the source.

Issue: Deployment failed; `ImportError: No module named 'requests.packages.urllib3'`

This is a common Fedora/RHEL/CentOS issue when certain system-required Python packages are managed using non-system tools (e.g. using `pip` or `setuptools` etc. rather than `yum`). In general, this can be resolved by the following commands:

```
sudo pip2 uninstall requests urllib3
sudo yum remove python-urllib3 python-requests
sudo yum install python-urllib3 python-requests
```

Issue: Changing ACSIA IP address after installation with clients connected

Please be aware that if you change the IP address of ACSIA in the configuration file this will break all communications with any connected clients. Attempting to amend this manually on the client side will not restore SSL/TLS sessions using the original keypair and therefore it will not function properly. Any accidental change to the ACSIA IP address can be fixed by running 'acsia_update_ip' from the ACSIA server itself.**

Issue: After update Kernel monitoring stop working on clients

It is not always the case but it may happen that during updating and patching activities on servers that are connected to ACSIA the Kernel level monitoring my not function properly. If you experience any issues with Kernel monitoring after updates all you need to do is to perform the following command as `acsia user on all connected clients:`cp -rf /etc/falco/falco.yaml.rpmnew /etc/falco/falco.yaml` and restart the Kernel module from ACSIA Web UI on clients.

Issue: Mariadb is failing to start after restart

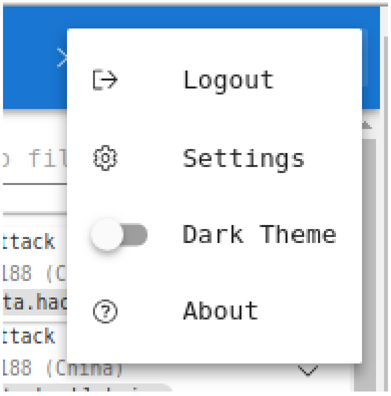
It is not expected to happen but sometimes during restart of ACSIA you may experience that one of the components of ACSIA such as `mariadb` will fail to start. In such circumstances please execute the following command as `acsia` user on ACSIA server itself `sudo -E docker-compose -f $ACSIA_ELK/docker-compose.yml restart mariadb`.

User Administration Section

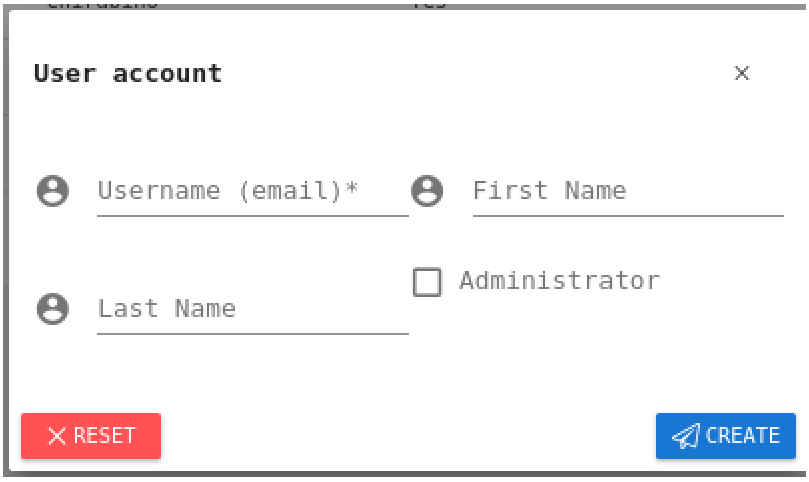
User administration section can be found at the top right bar by clicking on username with which you have logged in and then `Settings`.

Adding a New User

Adding a user never been simpler. Just click on `username` on top right bar and select `Settings` from the menu.



Then click on "ADD USER" and fill in all the fields, you can also Delete or Edit Users in this Section. Please keep in mind that the actual username has to be an email address.

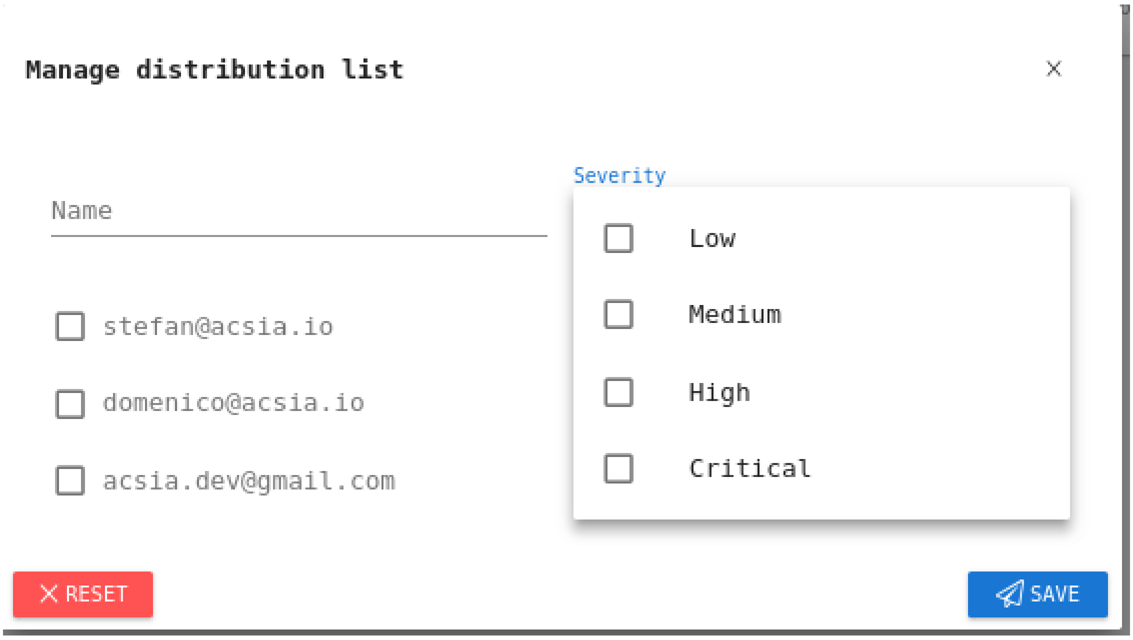


Distribution List

ACSIA enables you to create distribution lists where you can add members to each group and set the notifications to be sent to each distribution list. You can also set each distribution list to receive only `Critical` or `High` or `Medium/Low` priority security events. For instance, your CTO or CEO may not want to receive events outside `Critical` events and therefore a distribution list can be created to satisfy that need.

You will find the `Distribution List` on the left side bar menu. To create a new distribution list just click "ADD".

Give a name to the "Distribution List` created and select the users by adding them to the list along with choosing the type of event (Critical, High or Medium/Low) that you wish the group should receive.



You are now set up to receive notifications.

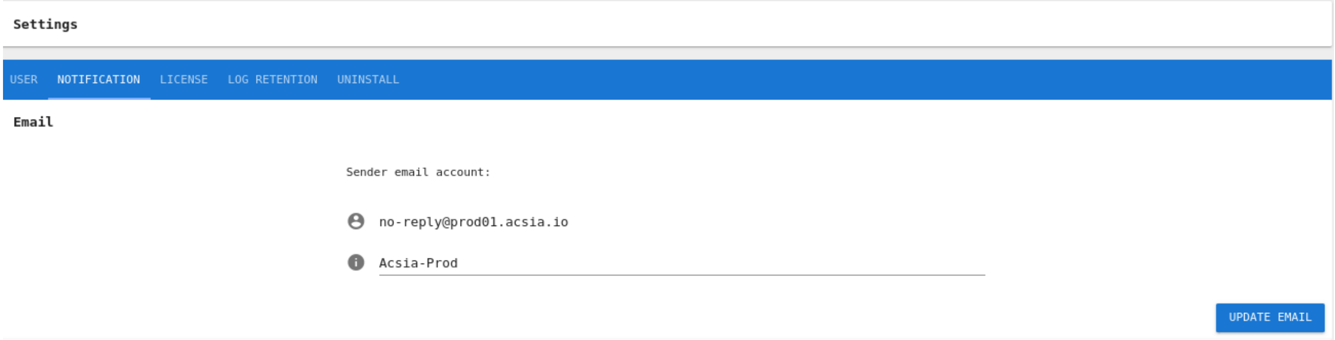
Email Settings

The email settings relate to ACSIA server side notification emails that notify about security events. This setting can be found at the top right bar, in the user menu under `Settings`.

Here you can set the sender email and the name for that email. For instance, if your organization domain is called `example.com` you can set the email as `no-reply@example.com` and the name as `Acsia Alerts` and white list that account on your anti-spam filters to make sure you receive notifications from that email account.

As soon as you set this up you will start receiving notification emails containing those sender parameters.

A screen-shot with an example of email settings is shown below.



Email

Sender email account:

no-reply@prod01.acsia.io

Acsia-Prod

Slack Integration

Similar to email notification mechanism, you can activate also notifications to be received through Slack.

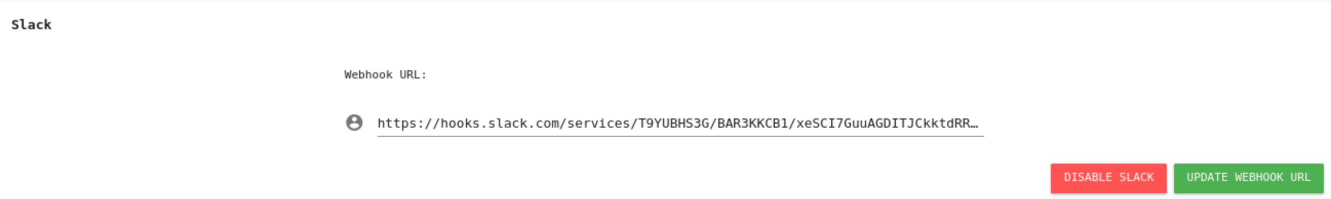
Slack setup instructions

Install incoming-webhook on your Slack

- Go to Apps
- Go to View App directory
- Search for `incoming-webhook`
- Go to Add Configuration
- At Post to Channel: choose channel/group where to send notification
- Click Add Incoming WebHooks Integration
- Go to Customize Name: Add name i.e. `Acsia Notifier`
- Copy the WebHook URL

Configure ACSIA to send notifications

- Access to the ACSIA UI
- Go to Settings -> Notification
- Activate Slack integration
- Paste the URL copied



Slack

Webhook URL:

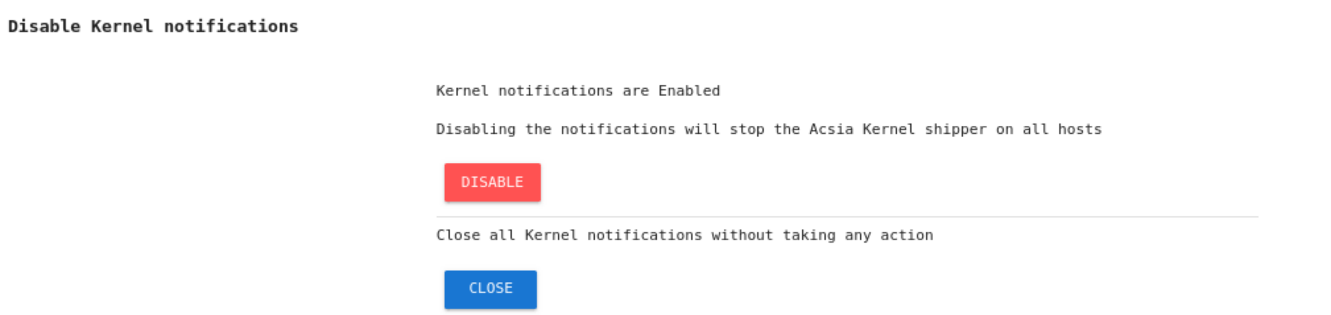
https://hooks.slack.com/services/T9YUBHS3G/BAR3KKCB1/xeSCI7GuuAGDITJCkktdRR…

Kernel Level Notifications

This is one of the modules that makes ACSIA unique in it's implementation. Thanks to it's kernel level monitoring, once is enabled, ACSIA have the ability to intercept the stream of every call made to kernel by intercepting the `syscalls` and searching for anomalies/threats in real-time.

If you'd like to receive kernel level notification it is recommended to keep this feature enabled. However, it can be disabled at anytime.

Below the snippet shows the kernel level notification enabled by default:



Disable Kernel notifications

Kernel notifications are Enabled

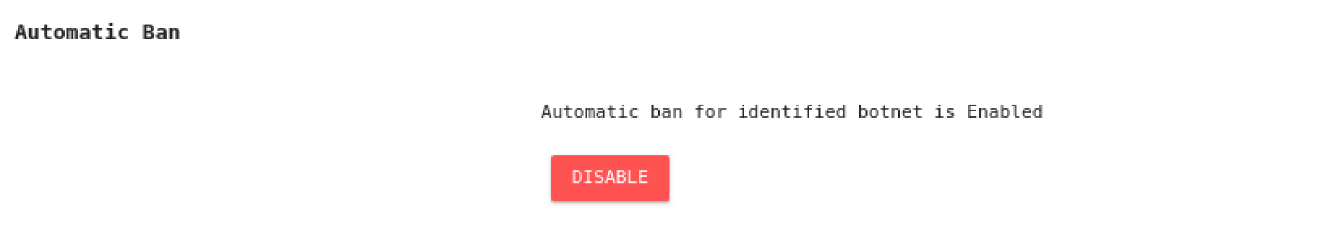
Disabling the notifications will stop the Acsia Kernel shipper on all hosts

Close all Kernel notifications without taking any action

Enabling Automatic Ban

Another ACSIA's unique feature is the automation of banning of the most common threats and attacks (i.e. BotNet attcks, etc.)

In the event one would like to stop automatically the most common attacks it is recommended enabling this feature.



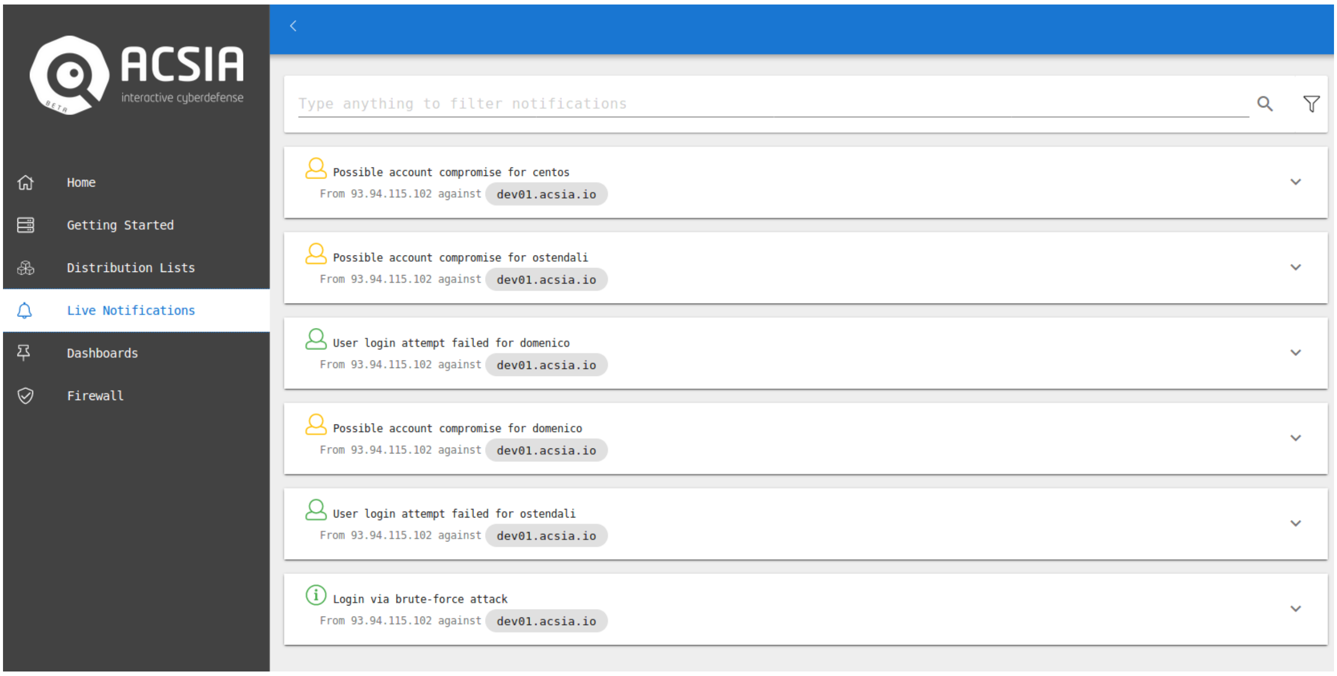
Automatic Ban

Automatic ban for identified botnet is Enabled

Live Notifications

On the left side menu we have also `Live Notifications` which contains the list of all live events that are not being actioned yet. All incoming security alerts will be listed in this screen and by clicking on `Details` button on each notification in this area you will be able to browse and explore the full details of the incident.

We also have filters where the events can be filtered based on `severity`, `event type`, `host` or also `keyword` based.



Possible account compromise for centos
From 93.94.115.102 against dev01.acsia.io

Possible account compromise for ostendali
From 93.94.115.102 against dev01.acsia.io

User login attempt failed for domenico
From 93.94.115.102 against dev01.acsia.io

Possible account compromise for domenico
From 93.94.115.102 against dev01.acsia.io

User login attempt failed for ostendali
From 93.94.115.102 against dev01.acsia.io

Login via brute-force attack
From 93.94.115.102 against dev01.acsia.io

Log Retention

ACSIA store all incoming logs from servers between Elasticsearch and MySQL databases. The lifespan of the logs can be set on ACSIA.

ACSIA enables user to set different retention for different type of logs:

Access Logs

These logs are usually includes all system logs.

Web Logs

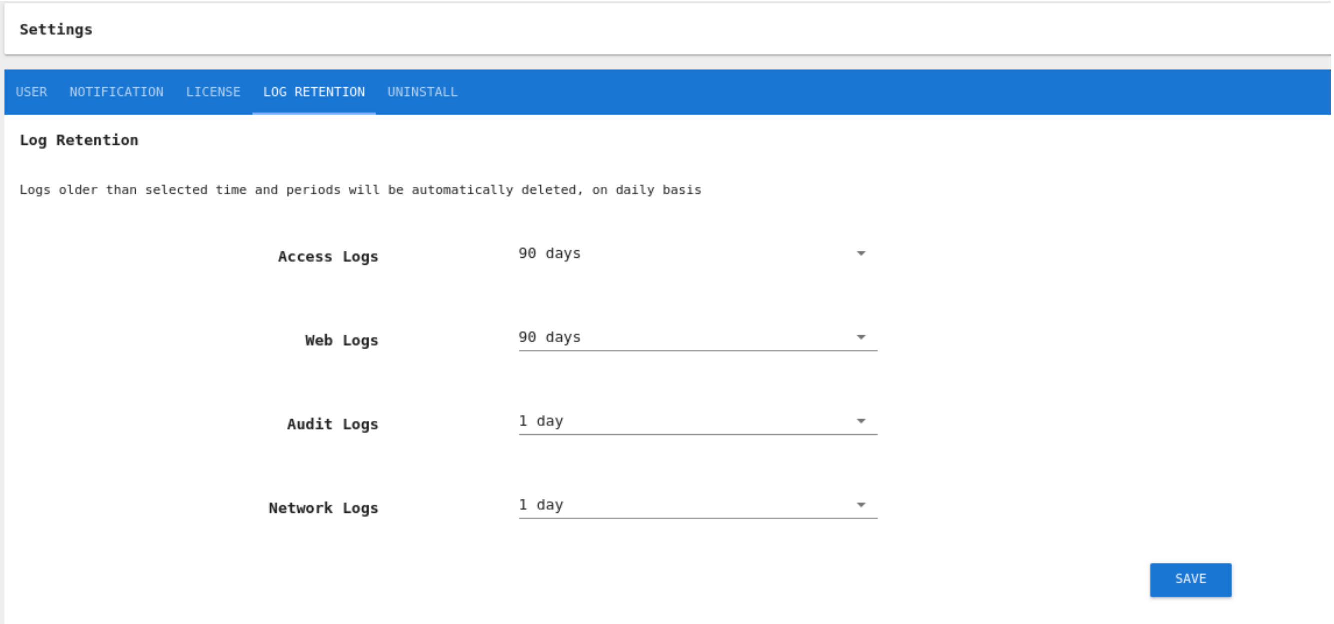
Web logs are those custom entered web application logs (i.e. apache, nginx, tomcat, etc.)

Audit Logs

These are very commonly known Linux audit logs.

Network Log

These logs are network traffic captured at server level.



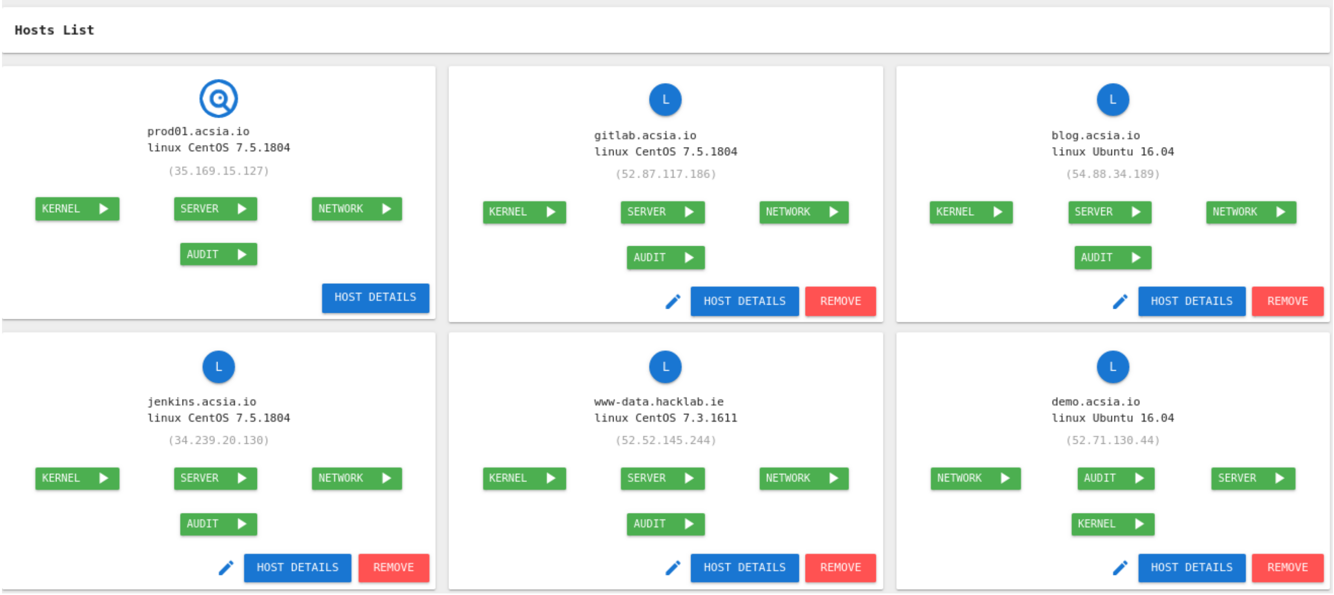
Log Retention

Logs older than selected time and periods will be automatically deleted, on daily basis

Access Logs	90 days
Web Logs	90 days
Audit Logs	1 day
Network Logs	1 day

Each individual log streams can be enabled/disabled at any time on each server from ACSIA UI Home page.

In the below screenshot we can see individual hosts having `Kernel`, `Server`, `Network` and `Audit` representing services for each log stream respectively. They can be stopped/started (enabled/disabled) by clicking on them.



Hosts List

prod01.acsia.io
linux CentOS 7.5.1804
(35.169.15.127)

gitlab.acsia.io
linux CentOS 7.5.1804
(52.87.117.186)

blog.acsia.io
linux Ubuntu 16.04
(54.88.34.189)

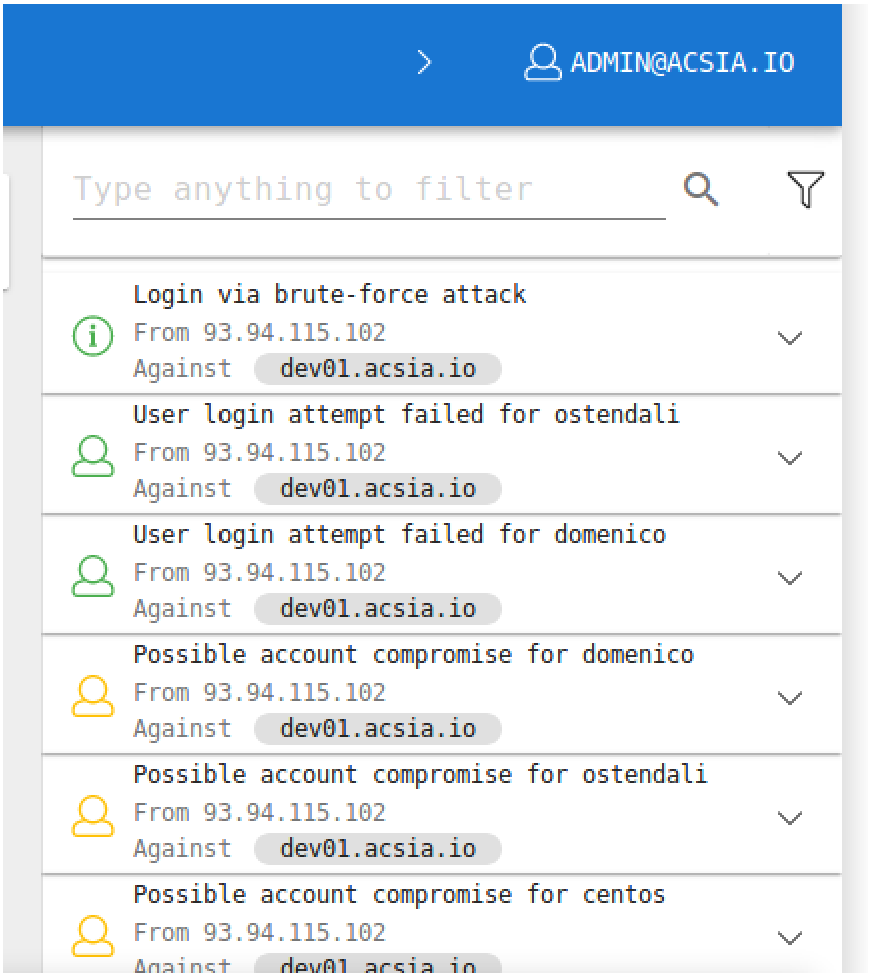
jenkins.acsia.io
linux CentOS 7.5.1804
(34.239.20.130)

www-data.hacklab.ie
linux CentOS 7.3.1611
(52.52.145.244)

demo.acsia.io
linux Ubuntu 16.04
(52.71.130.44)

Live Events Side Bar

We have a `Live Events` side bar on the right side of the screen. This side bar is similar to `Live Notifications` with an exception of being static and permanently on the right side handy to have an eye on events while browsing and surfing on ACSIA's other functionalities.



Login via brute-force attack
From 93.94.115.102
Against dev01.acsia.io

User login attempt failed for ostendali
From 93.94.115.102
Against dev01.acsia.io

User login attempt failed for domenico
From 93.94.115.102
Against dev01.acsia.io

Possible account compromise for domenico
From 93.94.115.102
Against dev01.acsia.io

Possible account compromise for ostendali
From 93.94.115.102
Against dev01.acsia.io

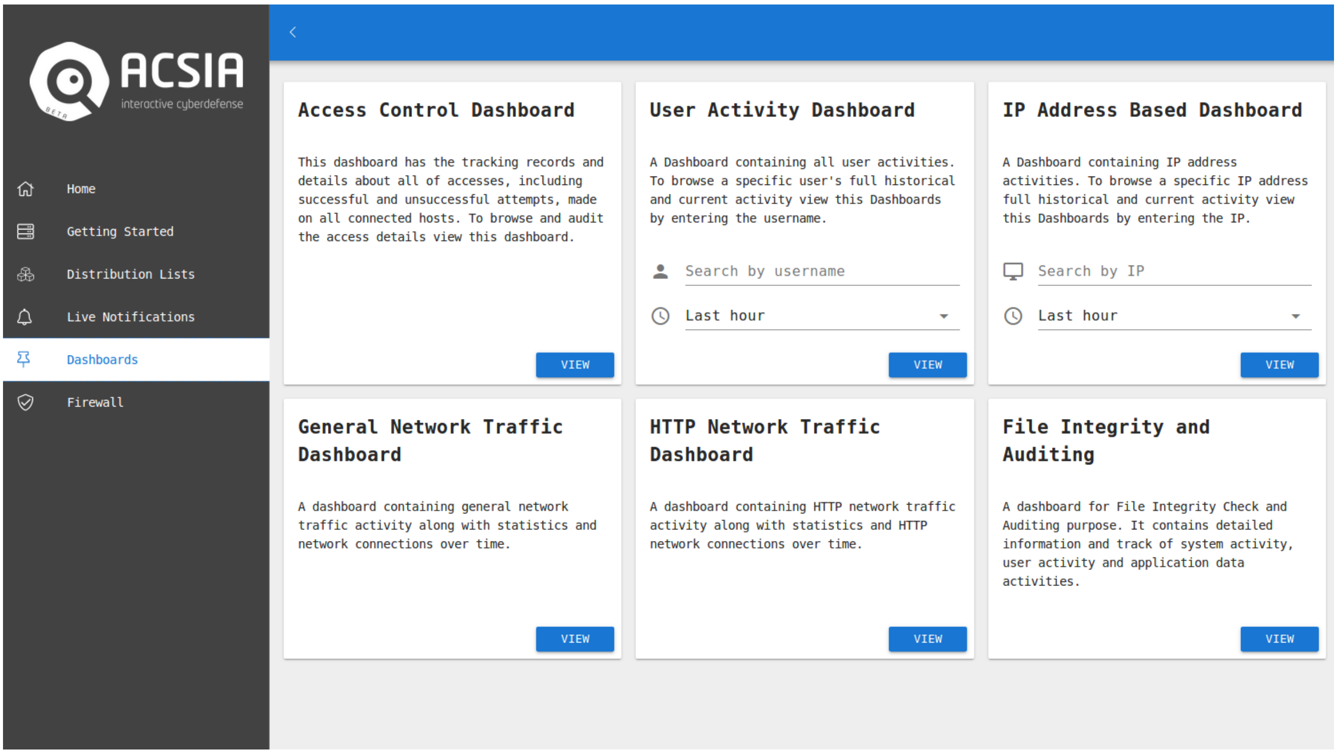
Possible account compromise for centos
From 93.94.115.102
Against dev01.acsia.io

Each event is shown with a pretty small summary but it comes embedded with `Immediate Actions` to enable the user to action and remediate an incoming security event or incident.

Dashboards

The `Dashboard` can also be found on the left hand side bar menu. This section contains multiple dashboards that ACSIA offer for deep investigations of events or even for generating reports and analytics.

Each dashboard is self-described as per the screen-shot shown.



Access Control Dashboard

This dashboard has the tracking records and details about all of accesses, including successful and unsuccessful attempts, made on all connected hosts. To browse and audit the access details view this dashboard.

User Activity Dashboard

A Dashboard containing all user activities. To browse a specific user's full historical and current activity view this Dashboards by entering the username.

Search by username
Last hour

IP Address Based Dashboard

A Dashboard containing IP address activities. To browse a specific IP address full historical and current activity view this Dashboards by entering the IP.

Search by IP
Last hour

General Network Traffic Dashboard

A dashboard containing general network traffic activity along with statistics and network connections over time.

HTTP Network Traffic Dashboard

A dashboard containing HTTP network traffic activity along with statistics and HTTP network connections over time.

File Integrity and Auditing

A dashboard for File Integrity Check and Auditing purpose. It contains detailed information and track of system activity, user activity and application data activities.

Firewall

ACSIA comes with its own embedded `Firewall`. This area is divided into 4 subsections as follows:

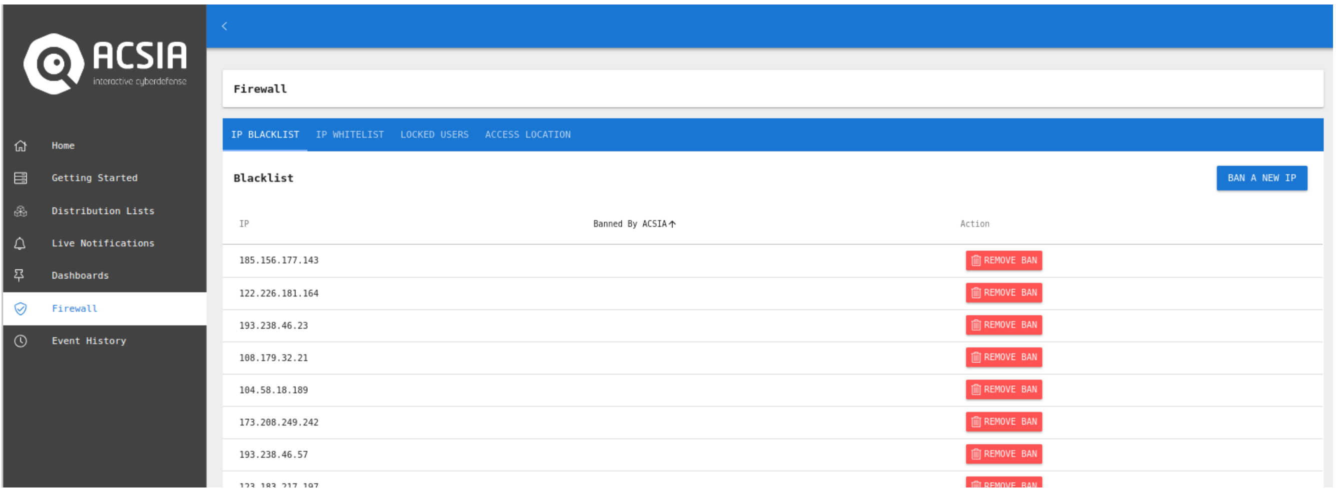
- IP Blacklist
- IP Whitelist
- Locked Users
- Access Location

The `IP Blacklist` is self-explanatory, it contains all those source IP addresses that been marked as malicious and unauthorized and therefore blacklisted. You always can undo and white list IP addresses in this area.

The `IP Whitelist` is self-explanatory, it contains all those source IP addresses that has been marked as **trusted**. You always can undo it so that the requests from that IP are analyzed. Note that white listing does not include web requests due to giving sensitivity of web level accesses. Therefore when you white list an IP address that doesn't apply to web requests.

The `Locked Users` contains specific users that are marked to be locked. They can be legitimate but attempted to non-authorized areas and waiting for clarification and investigation or they can be malicious users that are compromised the legitimate accounts details and therefore being locked. This actions can be also undo and users can be unlocked directly from here.

The `Access Location` refers to those security events where the access requests are originating from unauthorized physical locations. Therefore awaiting for approval or to be blacklisted. If you authorize a location based IP address for a user it is like whitelisting that user only for that IP address. In the other end, if you mark user unauthorized that user will still be able to access and make attempts but you will be notified. So it is something different than blacklisting unless you add to blacklist manually the IP or you lock the user itself.



Firewall

Blacklist

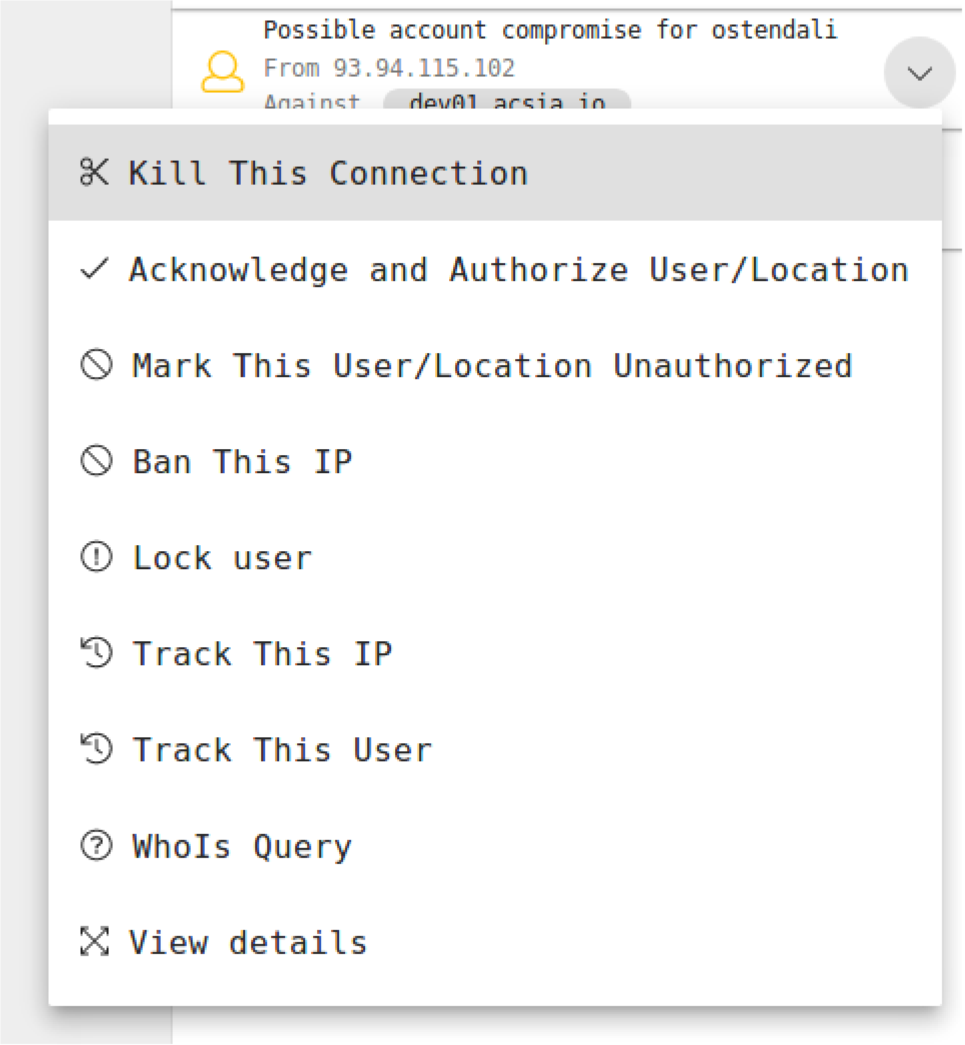
IP	Banned By ACSIA↑	Action
185.156.177.143		REMOVE BAN
122.226.181.164		REMOVE BAN
193.238.46.23		REMOVE BAN
108.179.32.21		REMOVE BAN
104.58.18.189		REMOVE BAN
173.208.249.242		REMOVE BAN
193.238.46.57		REMOVE BAN
123.183.217.197		REMOVE BAN

Immediate Actions

For the majority of clients, the `Immediate Actions` are probably going to be the most frequently used part of ACSIA. You will often find these actions embedded in all incoming security event or email notifications. From there you can take immediate action and interactively mitigate the events.

This is the `Interactive` feature of ACSIA:

The `Immediate Actions` supplied with the email notifications are ordered based on severity level of the event. For instance, if there is a potential account compromise, in the email notification the order priority algorithm will suggest what would be the best choice to take as first action and what would be next one and so on. The first action is placed in the first top order and also highlighted.



`Immediate Actions` are:

Kill this connection

By choosing this action you will be killing the actual connection and for the next 15mins incoming connections in inbound traffic for that IP address will be killed at first glance.

Acknowledge and Authorize User/Location

By choosing this action you authorize that specific user and the IP address on permanent (white list) basis to access your premises and that IP will be white listed for ACSIA.

Mark This User/Location as Unauthorized

By choosing this action you ask ACSIA to keep notifying about this event until you take a decision. Use this for incidents where you have not yet decided to ban or authorize.

Ban This IP

By choosing this action you ban permanently the IP, blacklist and therefore it will no longer be able to bother you.

Lock User

By choosing this action you lock the users account.

Track This IP

This action takes us to a `Dashboard` where the all network and server traffic will be populated for that specific IP so you will be able to check if that `IP` did or doing something else other than malicious attack and so on.

Track This User

This action takes us to a `Dashboard` where the all network and server traffic will be populated for that specific IP and enables us to check if that `user` did or is doing something else other than malicious attacks - and so on.

Whois Query

This is domain name lookup service to search the whois database for domain and IP registration information. It gives relevant information about the ownership of the originating IP address of the malicious user.

View Details

This provides accurate details of the event, including geographical location of the originating IP address and the location on the map and so on.

Close Incident

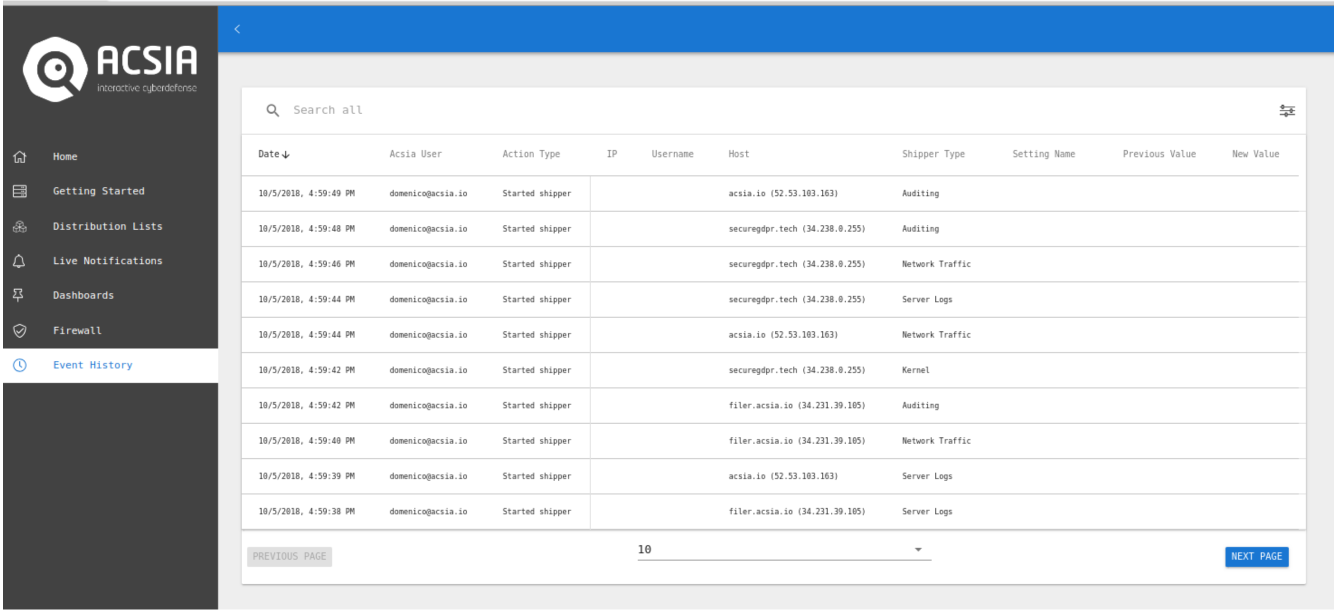
This is to simply disregard the event and therefore to let ACSIA notify you when it occurs again.

Find Sudo Session - Only for Linux clients

This is one of the most powerful and useful feature that comes with Kernel monitoring. From the moment that some suspicious activity have been detected or some user have attempted to read or write into sensitive data and files the alert will be triggered and within the alert you will have this option offered within immediate actions. When you click on this option within the alert, you will be presented with not only that specific action that triggered the alert but the entire session of that user in replay mode. Meaning, you will be able to view the full activity performed by user and therefore have an understanding why that user tried to alter the sensitive files and data and take action accordingly.

Event History

Event history is the section where you can find all events that are been actioned or amended and by whom.



Date ↓	Acsia User	Action Type	IP	Username	Host	Shipper Type	Setting Name	Previous Value	New Value
10/5/2018, 4:59:49 PM	domenico@acsia.io	Started shipper			acsia.io (52.53.103.163)	Auditing			
10/5/2018, 4:59:48 PM	domenico@acsia.io	Started shipper			securegdpr.tech (34.238.0.255)	Auditing			
10/5/2018, 4:59:46 PM	domenico@acsia.io	Started shipper			securegdpr.tech (34.238.0.255)	Network Traffic			
10/5/2018, 4:59:44 PM	domenico@acsia.io	Started shipper			securegdpr.tech (34.238.0.255)	Server Logs			
10/5/2018, 4:59:44 PM	domenico@acsia.io	Started shipper			acsia.io (52.53.103.163)	Network Traffic			
10/5/2018, 4:59:42 PM	domenico@acsia.io	Started shipper			securegdpr.tech (34.238.0.255)	Kernel			
10/5/2018, 4:59:42 PM	domenico@acsia.io	Started shipper			filer.acsia.io (34.231.39.105)	Auditing			
10/5/2018, 4:59:40 PM	domenico@acsia.io	Started shipper			filer.acsia.io (34.231.39.105)	Network Traffic			
10/5/2018, 4:59:39 PM	domenico@acsia.io	Started shipper			acsia.io (52.53.103.163)	Server Logs			
10/5/2018, 4:59:38 PM	domenico@acsia.io	Started shipper			filer.acsia.io (34.231.39.105)	Server Logs			

Email Notification

An example of email and Slack notification is shown below:

Potential account compromise is been detected. A user is accessing to your system (details below) is possibly not who claims to be.

Acsia Collected the following information

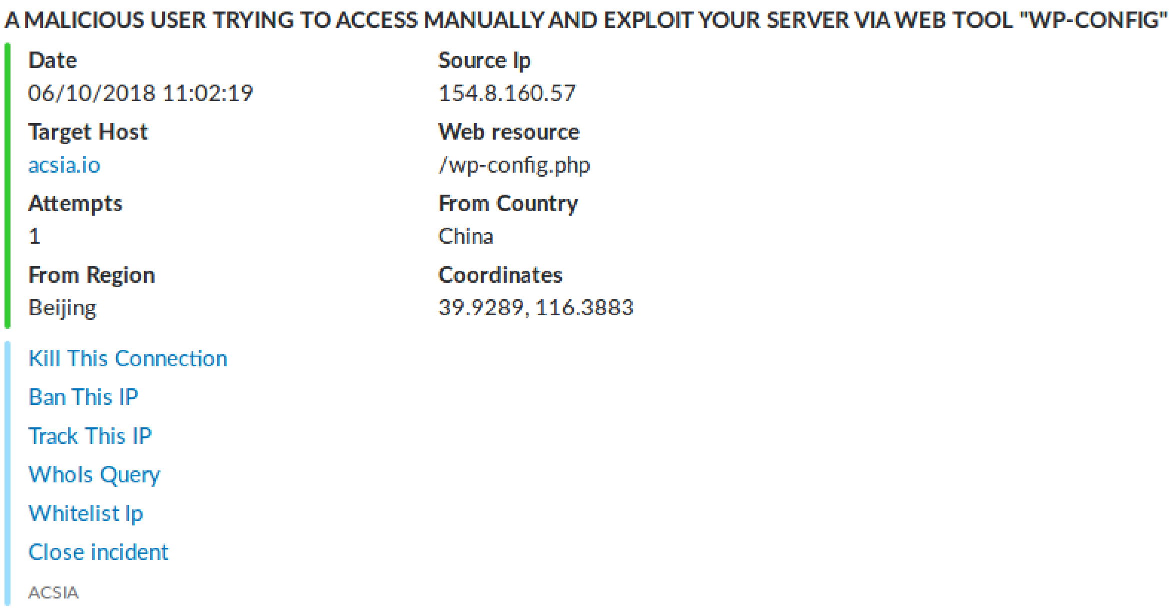
Date	Wed Feb 14 13:29:52 UTC 2018
Source IP	93.94.115.102
Target Host	preprod-acsia
Username	acsia
Authentication type	ssh key
Attempts	1

Source geographical location

Country	United Kingdom
Coordinates	51.4964, -0.1224

Immediate Actions

Kill This Connection	**Acknowledge and Authorize User/Location**
Mark This User/Location Unauthorized	**Ban This IP**
Lock user	**Track This IP**
Track This User	**WhoIs Query**



A MALICIOUS USER TRYING TO ACCESS MANUALLY AND EXPLOIT YOUR SERVER VIA WEB TOOL "WP-CONFIG"

Date 06/10/2018 11:02:19	**Source Ip** 154.8.160.57
Target Host acsia.io	**Web resource** /wp-config.php
Attempts 1	**From Country** China
From Region Beijing	**Coordinates** 39.9289, 116.3883

As we can see in the above screen-shot, we have received a notification about successful access using `acsia` user targeting our preproduction server. Now, we know that it is us who is logged in but ACSIA has legitimately notified this as potential account compromise.

This user and location will become legitimate for ACSIA only if we authorize the user and location as legitimate. ACSIA first needs to learn the legitimate accesses following our instructions. Therefore we need to indicate for the first time, using `Immediate Actions`, that this user is authorized and legitimate. From that moment onward ACSIA will know about that pattern and will no longer notify us(unless the account is really compromised one day).

We hope you are enjoying ACSIA!!!

For any further information and query please get in touch with our support team by contacting us via our web site (https://acsia.io) and be sure to visit our demo video and blog site too.

ACSIA is product of DKSU4Securitas Ltd.

W: 4securitas.com
W: acsia.io
B: blog.acsia.io
M: media.acsia.io