# OpenSecOps SOAR

## Installation Manual

# Table of Contents

## Document Versions

| Version | Date | Changes | Author |
|---|---|---|---|
| 1.2 | 2025-04-07 | Replaced “Delegat” with “OpenSecOps” throughout | Peter Bengtson |
| 1.1 | 2024-10-17 | Added section about support and SLAs | Peter Bengtson |
| 1.0 | 2024-09-25 | First version | Peter Bengtson |

# Introduction

This manual describes the OpenSecOps SOAR installation procedure.

# Audience

This installation manual is intended for an experienced system administrator with AWSAdministratorAccess permissions for all accounts in the system.

To you, an instruction like “Install Account Factory for Terraform (AFT)” is something you take in your stride. You should be intimately familiar with:

- Zsh
- The AWS CLI
- Git
- GitHub / GitLab / BitBucket
- CloudFormation
- Terraform, including common strategies for state files
- SAM, the AWS Serverless Application Model and its CLI
- Most AWS services

# Time Required

A very experienced system administrator should be able to install OpenSecOps SOAR in 2-3 days under ideal conditions. However, several factors could impact this timeline:

1. Familiarity with AWS services: The administrator would need to be very familiar with AWS Organizations, Security Hub, GuardDuty, IAM, and other related services.

2. Existing environment complexity: If installing into an existing AWS Organization, additional time might be needed for adapting the current setup.

3. Troubleshooting: Any issues that arise during installation could significantly extend the timeline.

4. Organisation-specific customizations: If extensive customisation is needed, this could add time to the process.

5. Approval processes: In larger organisations, obtaining necessary approvals for changes might extend the timeline.

6. Testing and verification: Thorough testing of all components after installation is crucial and could take additional time.

A more conservative estimate might be 3-5 days for most scenarios, allowing for potential complications and thorough testing. However, for a highly experienced administrator in a well-prepared environment with minimal customisation needs, 2-3 days is achievable.

# Prerequisites

- You must have an SSO user with full Administrator privileges in all accounts, including the administrative root account. Nothing must prevent your full access.
- Under no circumstances must the admin account root user be used for the installation.
- The system can be brand new, in which case there is only a single account, or it can be an existing system, potentially with a large number of member accounts. This installation guide is written primarily for the new installation scenario.
- The system can be installed in all regions.
- AWS Organizations is required. AWS Control Tower is optional.
- CLIs for OpenSecOps installer as described in https://github.com/OpenSecOps-AB/Installer/blob/main/README.md

# Account Tagging

Before installation, you should have thought about where you want email to be sent and where ServiceNow or Jira tickets should go. You can decide to divide these flows up into infrastructural and application destinations per account, or just use one endpoint for all.

Normally, all accounts are tagged individually in the root account where AWS Organizations resides. This information includes what environment type the account is, who is responsible for it, what team/squad/client is operating it, the IDs of ServiceNow or Jira boards to use, as well as email distribution list information.

If OpenSecOps Foundation is used, accounts will already be tagged as part of setting up Account Factory for Terraform (AFT) and the custom extensions provided by OpenSecOps Foundation.

If OpenSecOps Foundation is not used, the accounts must be tagged manually or through other automation.

Most information can default to a single endpoint, but in order to reap the full benefit of OpenSecOps SOAR the tag information should be as detailed and differentiated as you can make it per account. Any account not tagged will use the default ServiceNow/Jira board and email address, both defined during installation.

Before you proceed with installing OpenSecOps SOAR, you should have tagged all accounts in your organisation appropriately, as the first thing OpenSecOps SOAR will do once installed is to create issues directed towards responsible parties according to the information you have provided.

These are the tag names used by default in the OpenSecOps system, with their meaning:

| Configuration Parameter | Account Tag in AWS Organizations | Meaning |
|---|---|---|

| | | |
|---|---|---|
| AccountTeamEmailTag | `soar:account:distribution-list` | Contact email address |
| AccountTeamEmailTagApp | `soar:account:distribution-list:app` | Contact email address for the app domain (optional) |
| JiraProjectKeyTag | `soar:account:jira:project-key` | Jira Project Key |
| JiraProjectKeyTagApp | `soar:account:jira:project-key:app` | Jira Project Key for the app domain (optional) |
| ServiceNowProjectQueueTag | `soar:account:service-now:project-queue` | ServiceNow Project Queue |
| ServiceNowProjectQueueTagApp | `soar:account:service-now:project-queue:app` | ServiceNow Project Queue for the app domain (optional) |
| ProjectTag | `soar:account:project` | Project name (optional) |
| ClientTag | `soar:account:client` | Client name (optional) |
| TeamTag | `soar:account:team` | Team name |
| EnvironmentTag | `soar:account:environment` | Environment name |

Note that `soar:account:project` and `soar:account:client` are optional. You can rename them as you please and/or use them for other purposes. For instance, you could use the client tag to represent a squad, if your work mode involves squads.

# Create the Execution role in Org

In order for OpenSecOps SOAR to work in the Org account, we need to create the execution role OpenSecOps SOAR will use to do its work in all accounts. If you're using OpenSecOps Foundation and/or Control Tower, this is `AWSControlTowerExecution`, if you're not using Control Tower, it will be `OrganizationAccountAccessRole`.

This is in order so that this role can be assumed from its own account, just like for any other account in your Organization. We are thus creating parity in all accounts.

1. Log into the Org account
2. Go to IAM
3. Click on "Roles", then on "Create role"
4. Select "AWS Account"
5. Check that "This account" is selected. This is important. Only principals in the Org account should be able to assume the new role.
6. Click on "Next"
7. Find and select "AdministratorAccess".
8. Click on "Next"
9. Name the role "`AWSControlTowerExecution`" (if using Control Tower) or "`OrganizationAccountAccessRole`" (if not using Control Tower).

# Enable Service Managed Stack Sets

1. Log in to the Org account
2. Go to CloudFormation, then click on StackSets, then on Enable trusted access.
3. Go to AWS Organizations, then to Services and verify that trusted access has been enabled.

# Create the Security-Adm account

Create an account called “Security-Adm” if you don’t already have done so. OpenSecOps Foundation defines this account as part of its setup and places it in an OU called “Infra”. You may choose to do the same.

# Ticketing System

Now is the time to set up your ticketing system, if you want to use one. The alternatives are:

- JIRA Cloud
- ServiceNow

# Email domain

OpenSecOps SOAR uses email and tickets to notify stakeholders. The email domain must be a domain to which SES has rights to send. You may want to register a domain for this purpose using Route 53 and authorise it appropriately. We also recommend you to enable DKIM.

# Activations & OpenSecOpsions

## AWS Config

1. In the Org account, enable AWS Config in your main region. Remove the filter: in this region, you want to record IAM global events.
2. Enable AWS Config in your other enabled regions. Do not remove the IAM global filter in these regions.

## GuardDuty

1. Enable GuardDuty in all your activated regions.
2. OpenSecOpse administration of GuardDuty to the Security-Adm account in all your activated regions.
3. Log in to Security-Adm, enable and set up auto-enable in all your activated regions.

## IAM Access Analyzer

1. In Org, OpenSecOpse administration of IAM Access Analyzer to the Security-Adm account.
2. In Security-Adm, set up an organisation-wide IAM Access Analyzer for external access in all your regions.
3. Set up an organisation-wide IAM Access Analyzer for unused access in your main region only.

## Amazon Detective

1. OpenSecOpse Amazon Detective in all your regions to Security-Adm (the GUI will suggest this account automatically).
2. In Security-Adm, configure Detective in all your selected regions.

## Amazon Inspector

1. In the Org account, OpenSecOpse administration of Amazon Inspector to the Security-Adm account in all your chosen regions.

2. In the Security-Adm account, configure Amazon Inspector in each chosen region. Note that you must activate/invite the individual existing member accounts in each region as well as enable automatic activation of new accounts.

## AWS Security Hub

1. In the Org account, OpenSecOpse administration to the Security-Adm account as you enable Security Hub in all your enabled regions.

2. In the Security-Adm account, set up central configuration and consolidated findings in all your enabled regions.

3. Set up two policies, one for PROD and one for DEV accounts. Make sure that auto-enabling of new controls is *not* enabled. Also make sure that you select exactly the controls you need, one by one. The PROD policy will include things like multi-zone deployment, inclusion in backup plans, and deletion protection of resources. The DEV policy should not require these things, to ease development. Deleting a KMS key is a no-no in PROD, but should be allowed in DEV, for instance.
    a. Assign the PROD policy to the org root
    b. Assign the DEV policy to IndividualBusinessUsers, Sandbox, and SDLC OUs.

4. Suppress all findings in all regions to let the system start over with the new settings. Relevant findings will be regenerated within 24 hours. It's a good idea to wait 24 hours to verify your control setup.

## AWS Bedrock

In your Org account, in the region you wish to use for OpenSecOps SOAR's GenAI functionality, activate Anthropic Claude 3.7 Sonnet.

# Setting up CLI access to AWS

We will soon be setting up OpenSecOps Install, which is a command-line tool. It requires CLI access to your accounts. For this, you can use something like the following in your `~/.aws/config` file:

```
[default]
region = eu-north-1
sso_start_url = https://yourcompany.awsapps.com/start
sso_region = eu-north-1
sso_account_id = 111111111111
sso_role_name = AWSAdministratorAccess

[profile Org]
region = eu-north-1
sso_start_url = https://yourcompany.awsapps.com/start
sso_region = eu-north-1
sso_account_id = 111111111111
```

```
sso_role_name = AWSAdministratorAccess

[profile Log-Archive]
region = eu-north-1
sso_start_url = https://yourcompany.awsapps.com/start
sso_account_id = 222222222222
sso_role_name = AWSAdministratorAccess
sso_region = eu-north-1

[profile Audit]
region = eu-north-1
sso_start_url = https://yourcompany.awsapps.com/start
sso_account_id = 333333333333
sso_role_name = AWSAdministratorAccess
sso_region = eu-north-1

[profile AFT-Management]
region = eu-north-1
sso_start_url = https://yourcompany.awsapps.com/start
sso_region = eu-north-1
sso_account_id = 444444444444
sso_role_name = AWSAdministratorAccess

[profile Security-Adm]
region = eu-north-1
sso_start_url = https://yourcompany.awsapps.com/start
sso_region = eu-north-1
sso_account_id = 555555555555
sso_role_name = AWSAdministratorAccess
```

Update the `region`, the `sso_start_url`s, the `sso_account_id`s, and, if you wish, the `profile` names. Note that the `default` and the `Org` profile account details are the same; this is intentional.

# Setting up OpenSecOps Install

## Clone the repo

1. In your terminal, navigate to your `AWS` folder, then `cd` to `OpenSecOps-CLIENT-FOLDER`.
2. `git clone git@github.com:OpenSecOps-AB/Installer.git`

## Copy the configuration directory

1. `cd Installer`
2. `cp -rf apps.example apps`

## Set up source control for the configuration

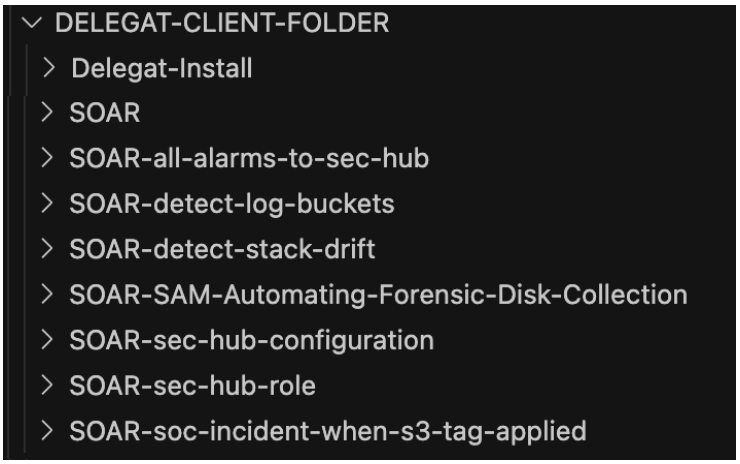
You will notice that the new `apps` folder is excluded from source control. As this folder will contain the entire configuration for OpenSecOps SOAR, it's *strongly* recommended that you create a Git repository for the `apps` folder in a central location, so that the configuration can be shared between cloud administrators.

You clearly know how to set this up, so please do it now before you start tailoring it to the client's installation.

## Download the Foundation repos

1. `cd` to `Installer`.
2. `./init SOAR`

Your OpenSecOps-CLIENT-FOLDER should now look like this:




## Configure Account IDs and SSO profile names

The first thing you need to modify is `Installer/apps/accounts.toml`. It looks like this:

```
# Account IDs and SSO profile names.
# NB: Enter the SSO profile name, not the account name.
# Below, they happen to be the same, which is quite OK.
# The profiles must have SystemAdministrator privileges.
#

admin-account.id              = "111111111111"
admin-account.profile         = "Org"

security-account.id           = "222222222222"
security-account.profile      = "Security-Adm"

log-archive-account.id        = "333333333333"
log-archive-account.profile   = "Log-Archive"

aft-management-account.id     = "444444444444"
aft-management-account.profile = "AFT-Management"

audit-account.id              = "555555555555"
audit-account.profile         = "Audit"
```

If you've named your profiles as per the example in "Setting up CLI access to AWS", you only need to update the account ids.

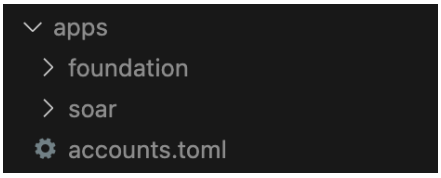
If you're not using AFT, simply leave `aft-management-account` as-is. Likewise if you're not using Control Tower: leave `log-archive-account` and `audit-account` as they are.

## Repo suppression

If you don't want to deploy a certain repo, make a copy of the `repos.toml` file and call it `repos-local.toml`. A file by that name will supersede `repos.toml`. Then set `deploy = false` for the repos you don't want to deploy.

## Configure OpenSecOps SOAR parameters

OpenSecOps SOAR, like OpenSecOps Foundation, is configured through text files. They each have their configuration folders, next to the account configuration file:




**Global Parameters**

The main configuration file is `parameters.toml`. First modify the first section:

```
# ------------------------------------------------------------
#
# Global parameters (used across repos)
#
# ------------------------------------------------------------

# AWS Organizations
org-id                      = 'o-xxxxxxxxxx'
root-ou                     = 'r-xxxx'

# Regions
main-region                 = 'eu-xxxxx-1'
other-regions               = ['us-xxxx-1']

# Either AWSControlTowerExecution (under Control Tower)
# or OrganizationAccountAccessRole (under plain AWS Organizations)
cross-account-role          = "AWSControlTowerExecution"

# The email domain to use for notifications
email-domain                = 'example.com'
```

You will find `org-id` and `root-ou` in AWS Organizations in your Org account.

The `main-region` should be set to your main region.

Set `other-regions` to the list of additional regions you have activated. If your main region is `eu-north-1` and your additional regions are `eu-central-1` and `us-east-1`, set this value to `["eu-central-1", "us-east-1"]`.

`email-domain` is the email domain from which OpenSecOps SOAR will send its emails. It must be a domain to which SES has rights to send. You may want to register a domain for this purpose using Route 53 and authorise it appropriately. We also recommend you to enable DKIM.

**SOAR section**

The next section you need to modify is the SOAR section. It begins like this:

```
# ------------------------------------------------------------
#
# SOAR
#
# ------------------------------------------------------------

[SOAR.SAM]

SOAREnabled            = 'Yes'                                  # Configure this
DeferIncidents         = "No"                                   # Configure this
DeferAutoRemediations  = "No"                                   # Configure this
DeferTeamFixes         = "No"                                   # Configure this

OrgAccountId           = "{admin-account}"
SecurityAccountId      = "{security-account}"
LogArchiveAccountId    = "{log-archive-account}"
AFTManagementAccountId = "{aft-management-account}"

CrossAccountRole       = "{cross-account-role}"
CustomEventBusName     = "{custom-event-bus-name}"

DiskForensicsInvoke    = "Yes"                                  # Configure this

MinAgeHours               = 24                                  # Configure this
ClearAccountDataCacheRate = "6 hours"                           # Configure this

# Mail ------------------------------------------------------------------------

SendEmail        = "Yes"                                        # Configure this
EmailCC          = "soc@example.com"                            # Configure this
EmailBCC         = ""                                           # Configure this

EmailSender      = "no-reply@{email-domain}"
EmailReturnPath  = "no-reply@{email-domain}"

# Ticketing -------------------------------------------------------------------

TicketingSystem                 = "JIRA"                    # Configure this
```

There are subsections for SOAR, Mail, Ticketing, Microsoft Sentinel, Tag keys, Overdue Ticket Limits, AI, Dashboards, and Environment Names. You only need to consider configuration items marked "`# Configure this`", but there's no need to

modify all of them. For a detailed description of each setting, refer to the OpenSecOps SOAR SOP and TDS.

Some tips:

- **SOAR subsection**:
  - You may want to set SOAREnabled, DeferIncidents, DeferAutoRemedations, and DeferTeamFixes all to "`Yes`" for the first 24 hours or so. This will have the effect of enabling the SOAR processing of events but disabling all user-facing management of issues. The only thing that will be done is suppressing ASFF events in your system for disabled Security Hub controls. When the SOAR has tidied all irrelevant events away af 24 hours or so, you can set DeferIncidents, DeferAutoRemedations, and DeferTeamFixes to "`No`" and redeploy.
  - DiskForensicsInvoke should be set to "No" if you're not using the Goldman Sachs disk forensics tool. Set it to "Yes" when you've generated the requisite AMIs and updated the relevant parameters.
- **Mail subsection**:
  - EmailCC should be set to a real email address or list of email addresses, or to "".
- **Ticketing subsection**:
  - TicketingSystem should be set to "None", "JIRA", or "ServiceNow".
- **Microsoft Sentinel subsection**:
  - If you want to send incidents to Microsoft Sentinel, configure this here.
- **Tag keys subsection**:
  - There's usually nothing to configure here.
- **Overdue Ticket Limits subsection**:
  - EscalationEmailCC should be set to a real address.
  - EscalationEmailSeverities: you may want to set this to "`CRITICAL,HIGH`" depending on company policy.
- **AI subsection**:
  - WeeklyReportEmailRecipients should be set to a real email address or list of email addresses
  - BedrockRegion should be set to the region in which you activated Bedrock and Anthropic Claude 3.5 Sonnet.
- **Dashboard subsection**:
  - You may wish to set DashboardName to "`CloudWatch-Default`" to make the OpenSecOps SOAR dashboard the CloudWatch default one.
- **Environment subsection**:
  - If the system uses environment names not in these lists, add them here.

**SOAR-SAM-Automating-Forensic-Disk-Collection**

The only other section you need to modify is SOAR-SAM-Automating-Forensic-Disk-Collection:

```
# ------------------------------------------------------------
#
# SOAR-SAM-Automating-Forensic-Disk-Collection
#
# ------------------------------------------------------------

[SOAR-SAM-Automating-Forensic-Disk-Collection.INFRA-diskForensicImageBuilder]

IAMRegion        = '{main-region}'
InstanceTypeList = 'm5.large,t3.large'


[SOAR-SAM-Automating-Forensic-Disk-Collection.SAM]

ORGID                              = '{org-id}'
ArtifactBucketName                 = 'delegat-soar-forensic-artifacts'
ArtifactBucketExpirationInDays     = 3650
ForensicsAMIs                      = "us-east-1: ami-000000000000, eu-west-1: ami-1111111111111"  # update this
AllowInvokeFromAccountId           = '{admin-account}'
TerminateRogueAfterInitialSnapshot = 'Yes'
InstanceType                       = 'm5.large'


[SOAR-SAM-Automating-Forensic-Disk-Collection.INFRA-diskMember]

MasterAccountNum = '{security-account}'
```

The only line you need to change is the value for ForensicsAMIs. If you're using this utility, enter the regions and names of the AMIs you've produced during post-installation. Then re-run the deployment. Don't forget to set DiskForensicsInvoke in the SOAR section to "`Yes`".

When installing for the first time, leave this section alone and set DiskForensicsInvoke to "`No`".

# Installing OpenSecOps SOAR

## Deployment

In a terminal, `cd` to your Installer folder.

Log into AWS using "`aws sso login`".

Before starting the installation, it's a good idea to check what the installation will do using "`./deploy-all SOAR --dry-run`". This will refresh the repositories, build the applications, upload the artifacts and inform you of what the installer will do.

If you encounter any problems, you might want to run the command again, but with the added switch "`--verbose`", thus: "`./deploy-all SOAR --dry-run --verbose`".

When you're ready, give the command "`./deploy-all SOAR`" and wait for completion.

## Post-Deployment

### Forensic Disk Collection

If you're using `SOAR-SAM-Automating-Forensic-Disk-Collection`, run the AMI pipelines in each region, take note of the AMI IDs and enter them in the

SOAR-SAM-Automating-Forensic-Disk-Collection section of the `parameters.toml` file. Set DiskForensicsInvoke to "Yes" and redeploy.

**Increase Concurrent Lambda Quota**

If the system is new, or if the Org account hasn't had much activity, you will need to request a quota increase for concurrent Lambda executions in that account, or SOAR processing will take longer than necessary. Use Service Quotas. It may be set to 10; request the maximum of 1000.

**Backup and Recovery**

You may wish to set up a backup and disaster recovery plan for the DynamoDB tables used by OpenSecOps SOAR. This can be done through any of the usual means, as documented by aws here:
https://aws.amazon.com/dynamodb/backup-restore/

You don't need to bother with backing up any of the S3 buckets installed, as they contain only transient data. The sole exception to this might be `infra-soar-soarweeklyaireportbucket-xxxxxxx` in the Org account, in case you want to back up the weekly security reports in HTML format.

# Updating and Recovering OpenSecOps SOAR

The procedure is described in the Installer README, at https://github.com/OpenSecOps-AB/Installer/blob/main/README.md.

Essentially, all you need to do is go to your Installer folder in a terminal and then:

```
1. git pull
2. ./deploy-all SOAR --dry-run
3. ./deploy-all SOAR
```

If you want to update a single component, go to its repo directory and do:

```
1. ./deploy --dry-run
2. ./deploy
```

It's generally prudent to do a dry run first to see what is going to change, but the dry run is of course completely optional.

NB: You can always delete all repositories, including the `Installer` one, provided you keep `Installer/apps` where all your configuration lives. You should already have set up source control for the `apps` directory. There is no configuration in the individual repositories.

If you delete the component repo directories, all you need to do is "`./init SOAR`" and fresh copies will be downloaded from OpenSecOps's central repositories.

# Technical Support & SLAs

We offer post-installation technical support and Service Level Agreements tailored to the client's needs and use case. For more information, please contact OpenSecOps AB's sales representatives.

# Conclusion

This concludes the installation of OpenSecOps SOAR.