



Teramind on AWS

Deployment Guide

Ver 1

Updated: 16/FEB/19



Table of Contents

Teramid on AWS Overview.....	2
Benefits of Deploying Teramid on AWS.....	2
Prerequisites	3
Step by Step Instructions.....	3
1. Creating an AWS Instance and Accessing it with SSH	3
2. Database Considerations	6
3. Creating the RDS Database	7
4. Storage Considerations.....	10
5. Creating an IAM Role for the S3 Storage	10
6. Finishing the Deployment	12
Architecture.....	16
Technical Specifications, Licensing and Support	17

Teramid on AWS Overview

All Teramid solutions are available to deploy on Amazon’s distributed, highly scalable and reliable Amazon Web Services (AWS) platform as a Private Cloud option. This deployment guide will help you discover what you can expect from your Teramid on AWS deployment and provide you with installation prerequisites, step by step instructions, technical and support information.

Benefits of Deploying Teramid on AWS

If you prefer on-premise deployments but unwilling to incur the cost and hassle of an in-house infrastructure or you want to move to the Cloud but have been concerned about compliance, security or have made the transition and already host on AWS, then Teramid on AWS is the right choice for you. With our AWS Private Cloud hosting option, enjoy the benefits of both worlds: cost and scalability comparable to a Cloud deployment and security and flexibility similar to an On-Premise solution. Here are some infrastructure benefits you can expect if you choose to deploy Teramid on AWS:



Flexible and Competitive Pricing

You only pay for the AWS resources you consume (i.e. CPU, storage, memory). Teramid's recommended configuration for a standard deployment (m4.xlarge instance, up to 100 users) costs only \$0.2 per/hr.



Optimized Environment

Choose the region, OS, storage, database etc. For example, you can use an Amazon S3 storage for Teramid’s session recordings and Amazon RDS databases for session logs that’s highly optimized for performance on AWS.



Easy Deployment

Create a Teramid server with a single click by launching the Teramid AMI (Amazon Machine Image) from the AWS Marketplace.



Reliability

Support for High Availability (HA), redundancy with multi-geo replications, on-demand backup and disaster recovery.



Scaling

Vertical and horizontal scaling with optional auto scaling that adjusts capacity based on demand.



Central Management Console

Configure and manage all your deployments from one central location.



Security and Compliance

Firewall, encryption at rest, SSL encryption, VLAN, SSH tunnels, 2FA, IP whitelisting and encrypted disks allowing for easy regulatory compliance for HIPAA, GDPR, PCI DSS and more make it ideal for Teramid customers in government, healthcare, finance and other regulated industries.

Prerequisites

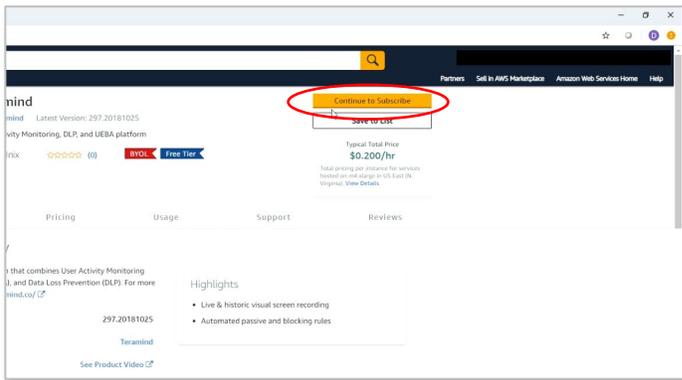
To get started, you will need:

- An AWS account
- Your Teramind license key, available from Teramind Self-Hosted portal at: <https://www.teramind.co/portal>
- An SSH client like Putty if you are using Windows

Step by Step Instructions

Teramind's Amazon Marketplace portal makes it very easy to get started with Teramind on AWS. Just choose a region closest to you, pick an instance type, check the estimated infrastructure cost and then deploy the instance(s) in minutes with a ready-made Amazon Machine Image (AMI). Below are the step by step instructions for deploying Teramind on AWS.

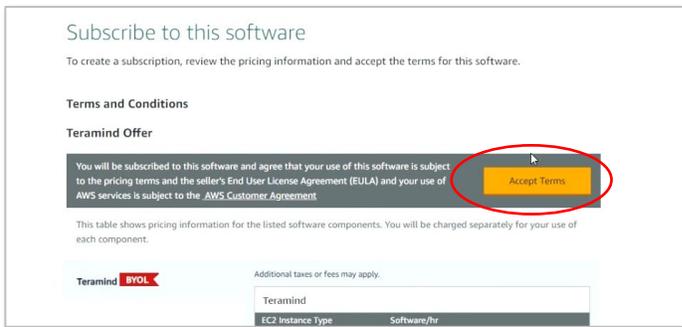
1 Creating an AWS Instance and Accessing it with SSH



Step 1-1

Visit: <https://www.teramind.co/deployment/aws> and click the **Check out Teramind on Amazon Marketplace** button, which will take you to Teramind’s deployment page on AWS.

Once there, click the **Continue to Subscribe** button on the top right corner.



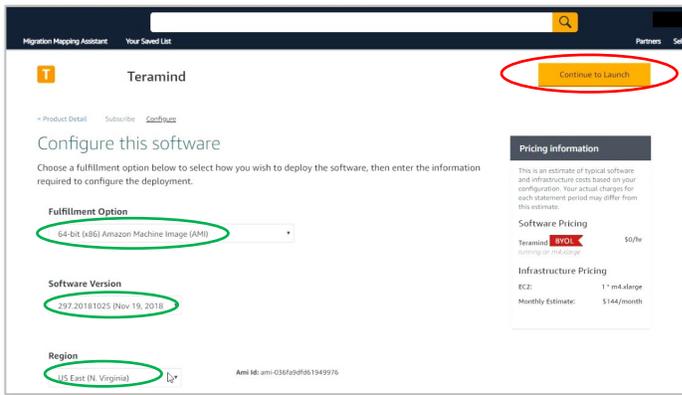
Step 1-2

On the 'Subscribe to this software' screen, click the **Accept Terms** button.



Step 1-3

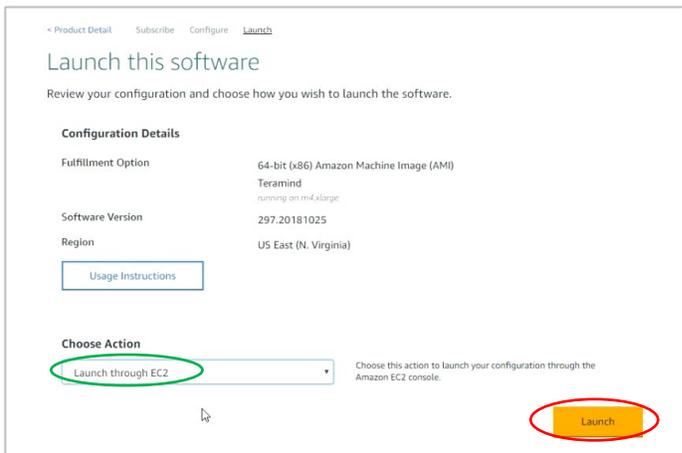
Click the **Continue to Configuration** button.



Step 1-4

On the 'Configure this software' screen, you can choose the **Fulfillment Option**, **Software Version** and the **Region**. Be sure to select a region closest to you for better performance.

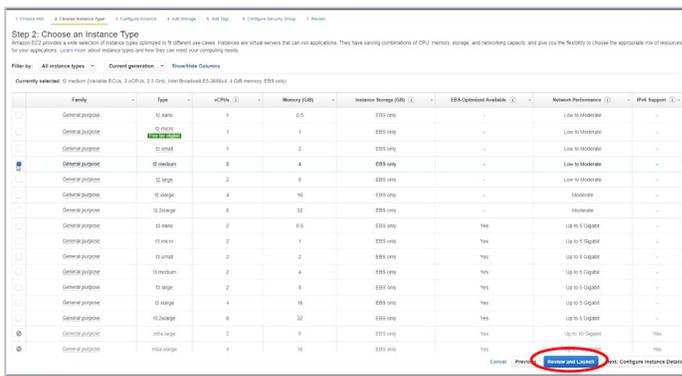
Click the **Continue to Launch** button when done.



Step 1-5

On the 'Launch this software' page, click on the **Choose Action** pull-down menu and select **Launch through EC2**.

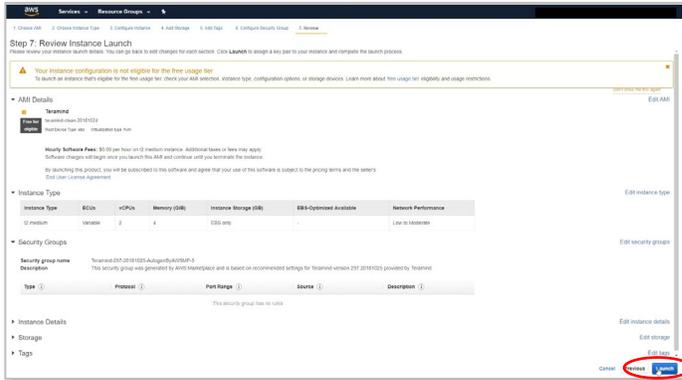
Click the **Launch** button.



Step 1-6

On the 'Choose an Instance Type' screen, select an instance type from the list of available instances. *Teramind recommends at least an m4.xlarge for a typical deployment of up to 300 users.*

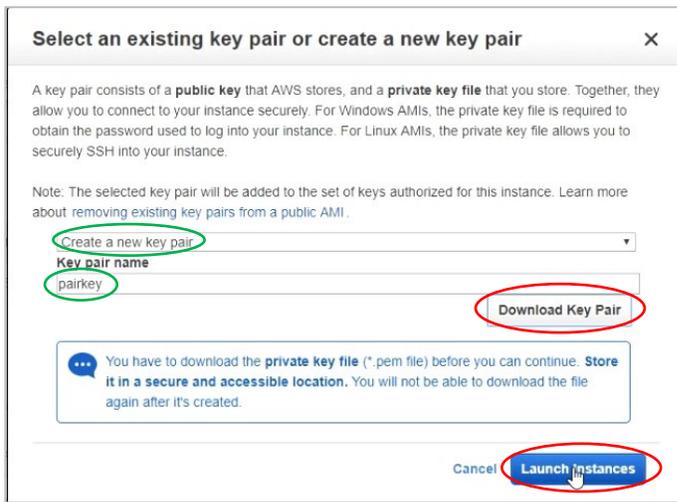
Once done, click the **Review and Launch** button at the bottom of the page.



Step 1-7

The 'Review Instance Launch' screen shows all the details associated with your selected instance. You can edit any configuration if needed.

Click **Launch** to accept the configurations and launch the instance.



Step 1-8

Once the instance is ready, a window will pop-up where you can select an existing key pair or create a new pair.

From the first pull-down menu, select **Create a new key pair** and give it a **name**. Click the **Download Key Pair** button and save the file in a secure place.

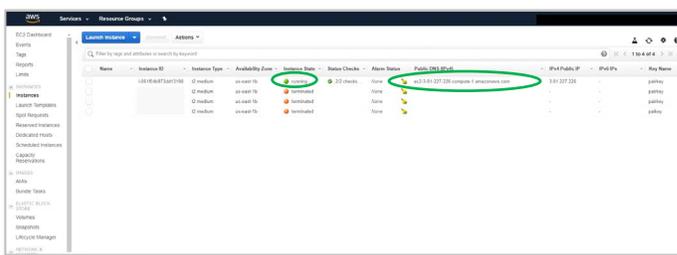
Once done saving, click **Launch Instances**.

The instance will be initialized in a few moments



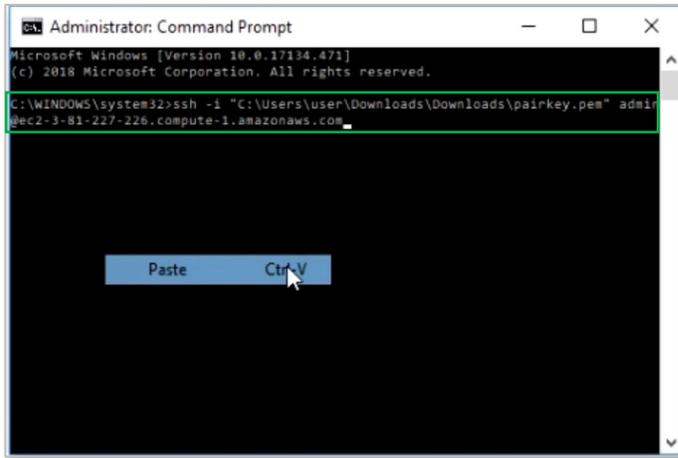
Step 1-9

On the Launch Status window, click the **View Instances** button at the bottom to see your newly created instance(s).



Step 1-10

When the Instance State shows **running**, it means it's now ready for accepting connections. Copy the **Public DNS (IPv4)** address for the instance you just created and copy it or write it down.

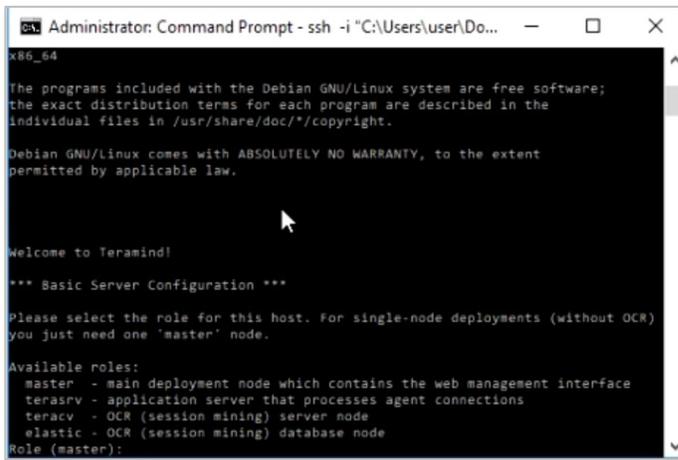


Step 1-11

Launch a SSH session. If you are on Windows, you can use a tool like Putty or a similar utility for the SSH. Make sure you have administrative access.

Type: `ssh -i "path/to/your/key-pair.pem" admin@`

Paste/type the **Public DNS** you copied in the previous step just after the @ symbol in the command prompt. Press Enter. The server will be ready in few minutes.



Step 1-12

Once the server is ready, you will see this message, 'Welcome to Teramind!'. Under the message you will be given options to assign roles for the nodes/hosts, beginning with the Master role. However, before you can assign a role, you will need to finish the rest of the steps.

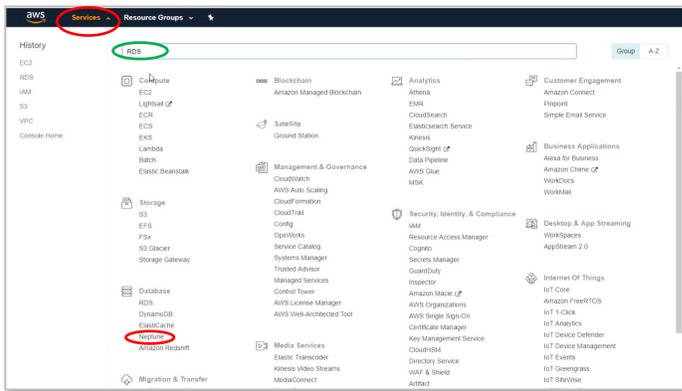
Keep this command window open as you will use it to finalize the setup at a later stage.

2 Database Considerations

Before you proceed any further with the setup process, you will need to determine if you want to use external databases or storage with the instance. External databases or storage aren't mandatory. However, you may want to use Amazon RDS as the database, and Amazon S3 for screen and audio recordings. Using RDS and S3 will improve the scalability of your platform and is recommended for deployments over 100 concurrent users.

This guide will show you how to use external database and storage as an example using an AWS RDS database and the AWS S3 storage service. First, it will show you how to create an RDS database. If you already know how to create an RDS database, you can skip the next section.

3 Creating the RDS Database

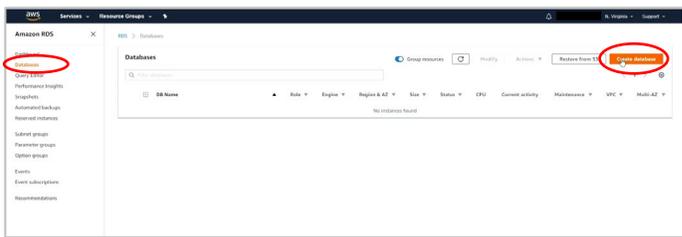


Step 3-1

From your EC2 Management Console, click **Services** from the top left hand side. A Few options will show up.

Type **RDS** in the search field.

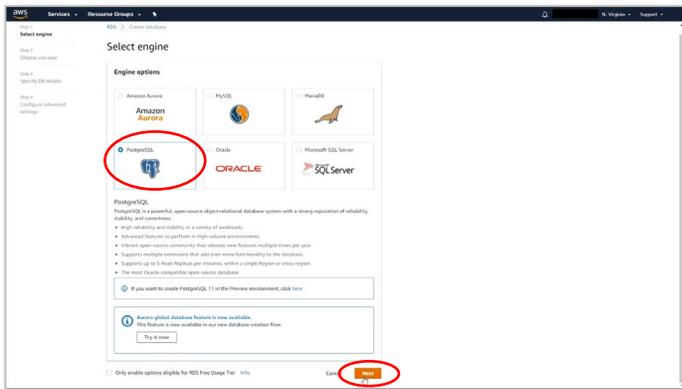
Select the **RDS** under the Database category.



Step 3-2

Select **Databases** from the left-hand sidebar.

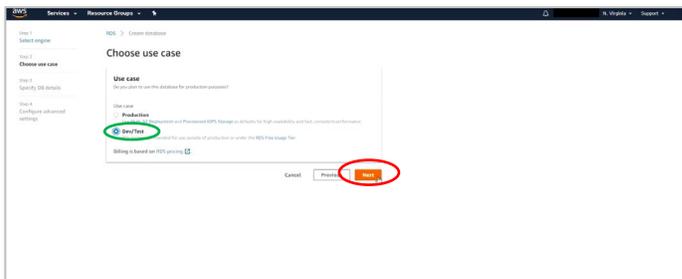
The, click the **Create database** button on the top-right corner of the screen.



Step 3-3

On the 'Select engine' screen, select **PostgreSQL**.

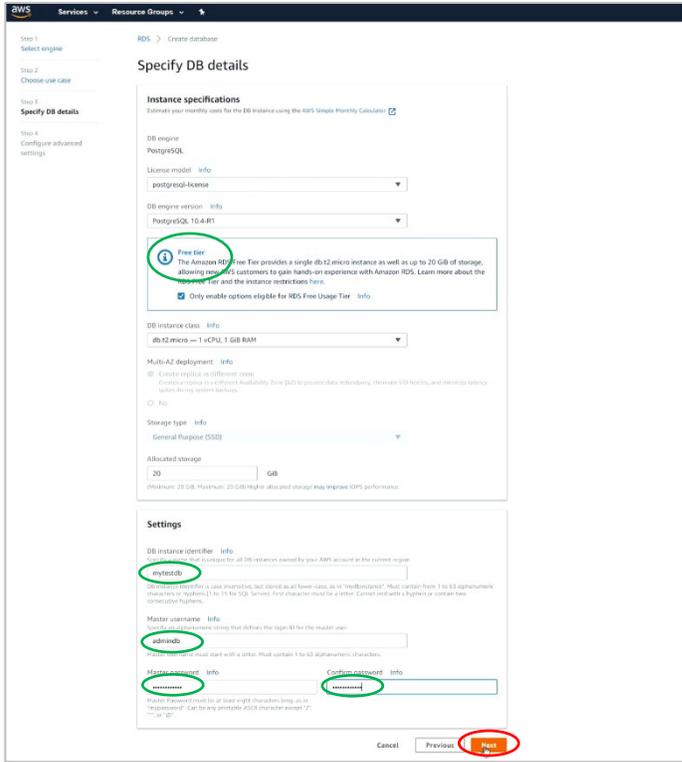
Click **Next** to go to the next screen.



Step 3-4

On the 'Choose use case' screen, select either **Production** or **Dev/Test** depending on what you want to do. In this case, we will use the Dev/Test option.

Click **Next** to go to the next screen.



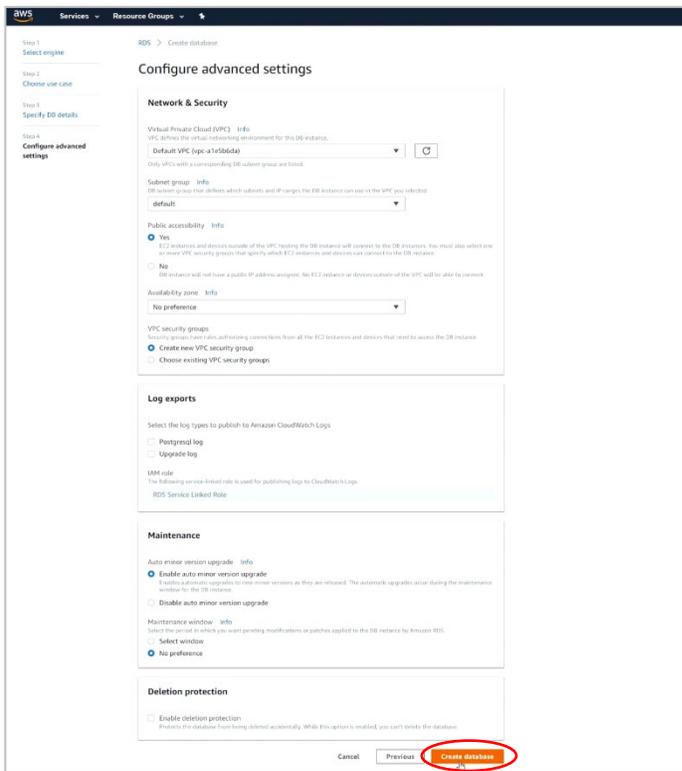
Step 3-5

On the 'Specify DB details' screen, you will notice that there's a **Free Tier** option. You can enable this if you are running test databases and don't want to pay for its usage.

Scroll down until you see the 'Settings' section.

Type a name in the **DB instance identifier** field. Assign a **Master user name** and a **password**.

When done, click the **Next** button.

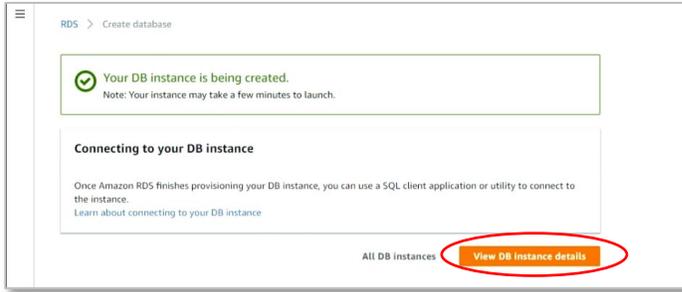


Step 3-6

The 'Configure advanced settings' screen has options to configure Network & Security, Log exports, encryption, backup etc. You can adjust them as needed.

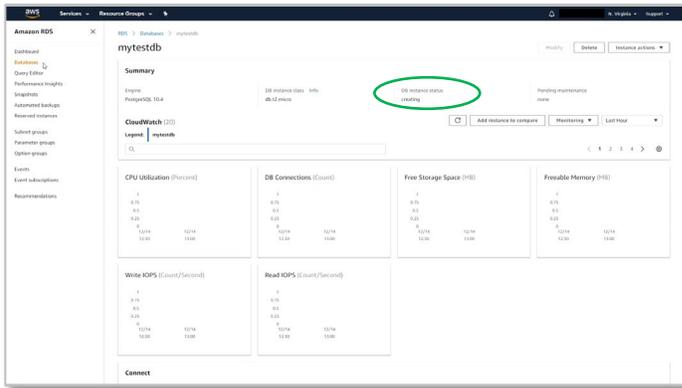
In this guide, we will leave them to their default settings.

Click the **Create database** button.



Step 3-7

Once the instance is created you will see a **View DB instance details** button. Click it.



Step 3-8

On your database 'Summary' screen, you will see the details like the Engine, DB instance class, CPU Utilization, DB Connections, Free Storage Space etc.

When the **DB instance status** changes to 'Available', the database will be ready for use.

4 Storage Considerations

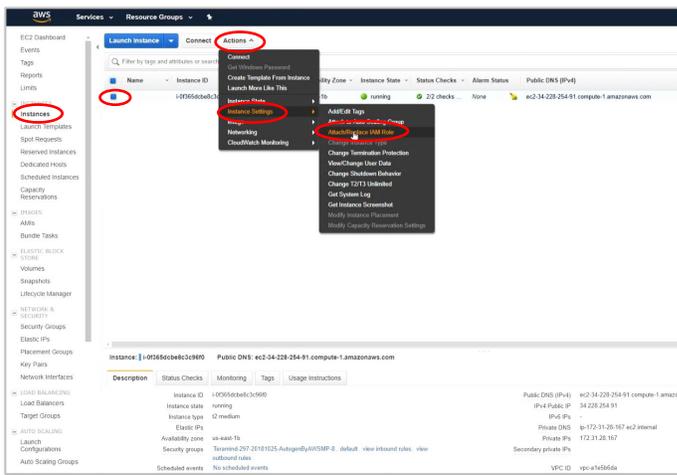
Teramind doesn't need external storage. However, Teramind can utilize a file storage system for screen and audio recordings, captured attachments, printer documents, and metadata such as user images. AWS S3 is an efficient, secure, scalable and redundant solution to store objects like these.

The S3 storage will improve the scalability of your platform and is recommended for deployments of over 100 concurrent monitored users.

5 Creating an IAM Role for the S3 Storage

To use the AWS S3 storage, you will need an Identity and Access Management or IAM profile. IAM enables you to manage access to your AWS services and resources securely. Using IAM, you can create and manage AWS users and groups and control user permissions.

This guide will show you how to create an IAM profile. If you already know how to create one, you can skip this section.

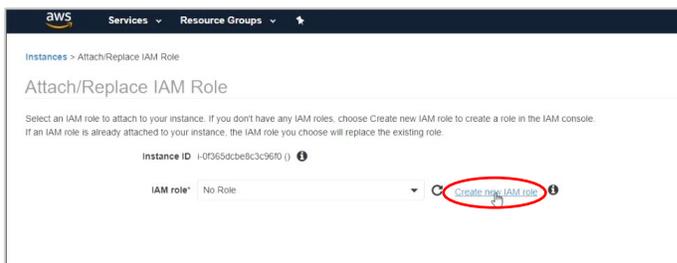


Step 5-1

Go to the **Instances** screen from your EC2 Management Console. Make sure the instance you want to use is selected.

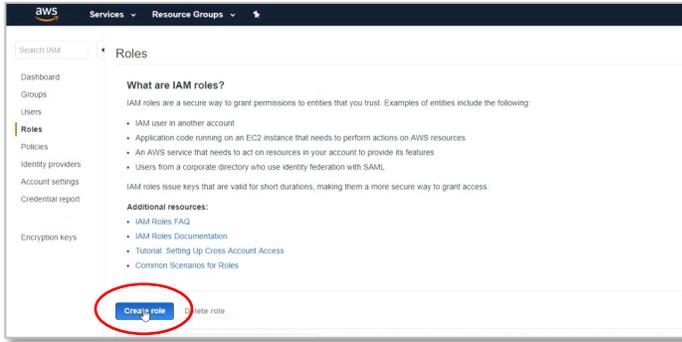
Click the **Actions** button on top of the instance names.

From the menu, select **Instance Settings > Attach/Replace IAM Role**.



Step 5-2

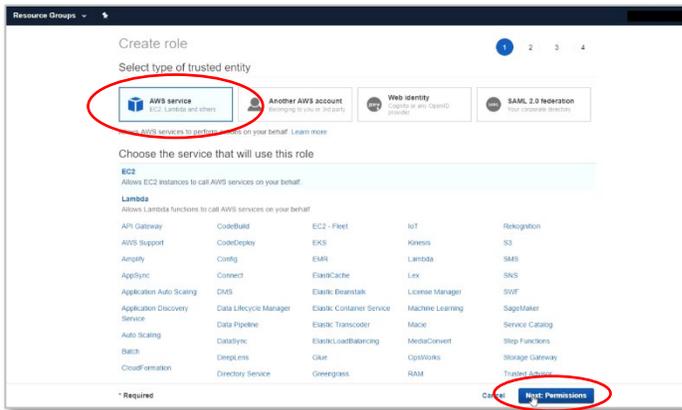
On the 'Attach/Replace IAM Role' screen, click the **Create new IAM role** link.



Step 5-3

On the 'Roles' screen you will see an explanation of the IAM roles and links to some resources and documentations.

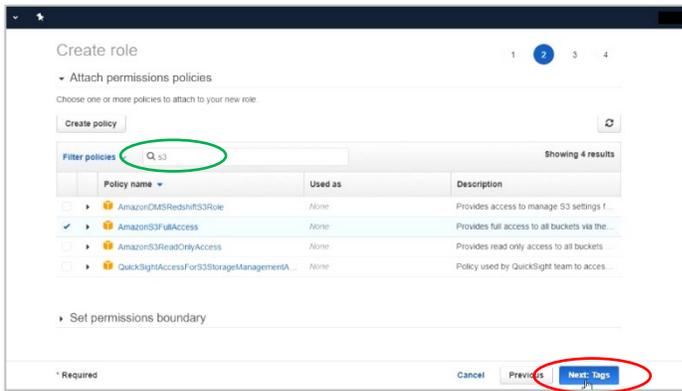
Click the **Create role** button to continue.



Step 5-4

Select the default **AWS service** as the type for trusted entity.

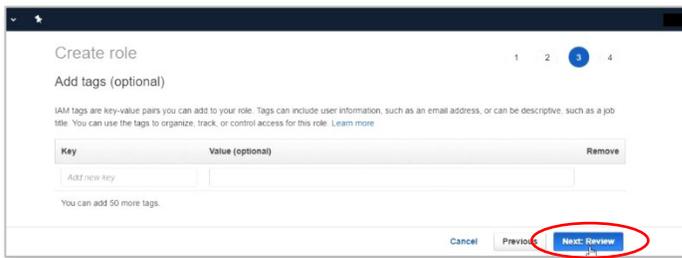
Click the **Next: Permissions** button.



Step 5-5

From the list of permission policies, select the one named '**AmazonS3FullAccess**' (you can quickly locate it using the Search field).

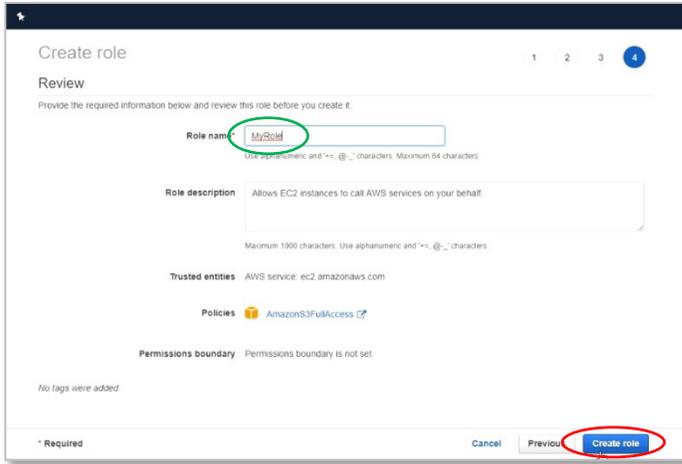
Click the **Next: Tags** button to continue.



Step 5-6

Tags are optional and we will not use any in this guide. You can still add some tags if you want.

Click the **Next: Review** button to go to the next screen.

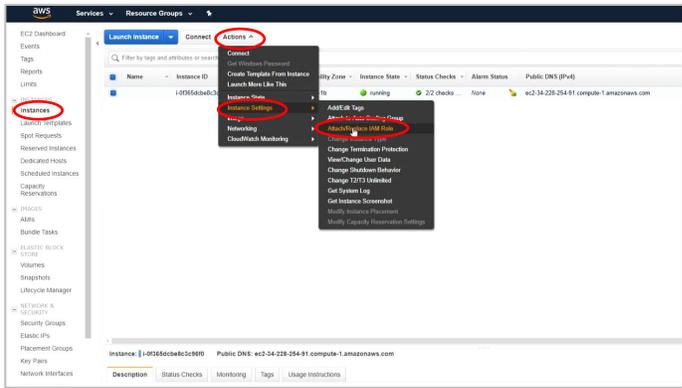


Step 5-7

On the 'Review' screen, give the role a **name** and click the **Create role** button.

6 Finishing the Deployment

Now that you have created the RDS database and an IAM profile for the S3 storage, you are now ready to continue with the rest of the deployment process.

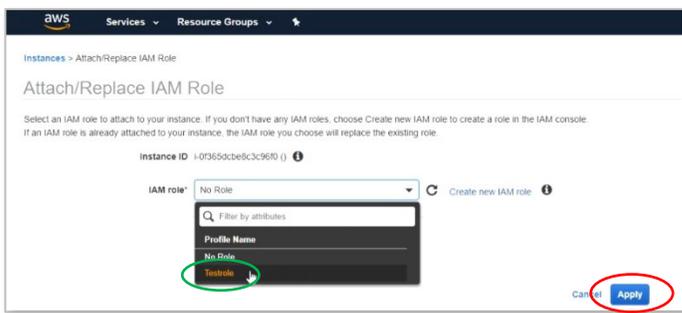


Step 6-1

Go to the **Instances** screen from your EC2 Management Console. Make sure the instance you want to use is selected.

Click the **Actions** button on top of the instance names.

From the menu, select **Instance Settings > Attach/Replace IAM Role**.



Step 6-2

On the 'Attach/Replace IAM Role' screen, click the **IAM role** pull-down menu and select the IAM you created in the previous steps.

Click **Apply** to assign the IAM to the selected instance.

```

Administrator: Command Prompt - ssh -i "C:\Users\user\Do...
master - main deployment node which contains the web management interface
terasrv - application server that processes agent connections
teracy - OCR (session mining) server node
elastic - OCR (session mining) database node
Role (master): master

Teramind comes with a pre-installed PostgreSQL database server.
You can also choose to use a dedicated database host (ie. Amazon RDS,
Azure DB for PostgreSQL, ...). If you prefer to use an external DB
server, you'll need to have the hostname, username, password ready.

Use external database (yes/NO): yes

*** NOTE ***
Teramind requires two databases named 'teramind' and 'ta_onsite'.
If those databases do not exist, the installation process will
attempt to create them. Therefore, CREATEDB permissions are
necessary.
***

Database IP or hostname: teramindnewdb.c1fbozwh3pd.us-east-1.rds.amazonaws.com
Database username: admindb
Database password:

```

Step 6-3

Go back to the command prompt window you used in 'Step 1-12'.

Give the master role a **name**.

On the 'use external database?' prompt enter **Yes**.

Supply the RDS **database IP / hostname**, **user name** and **password** you created before. Press enter when done.

Teramind will initialize the database, tables and required indexes automatically.

```

Administrator: Command Prompt - ssh -i "C:\Users\user\Do...
Teramind can store screen recordings and other non-meta-data as flat files or
as objects in a S3-compatible bucket.

Use S3 for storing data (yes/NO): yes

*** NOTE ***
This host should have IAM role for accessing S3.
***

Teramind can use existing S3 buckets or create new ones.

Create new buckets (yes/no): yes
Enter bucket name for screen recordings: teramindbuckettest1
Enter bucket name for user content (ie. attachments, printed documents, ...): te
ramindbuckettest2
Enter bucket name for Teramind application objects: teramindbuckettest1
Starting Teramind

Created symlink /etc/systemd/system/multi-user.target.wants/teramind.service +
/etc/systemd/system/teramind.service.

Teramind initial configuration is complete. Please open https://54.89.185.179/
n
a browser to continue system setup.
admin@ip-172-31-28-167:~$

```

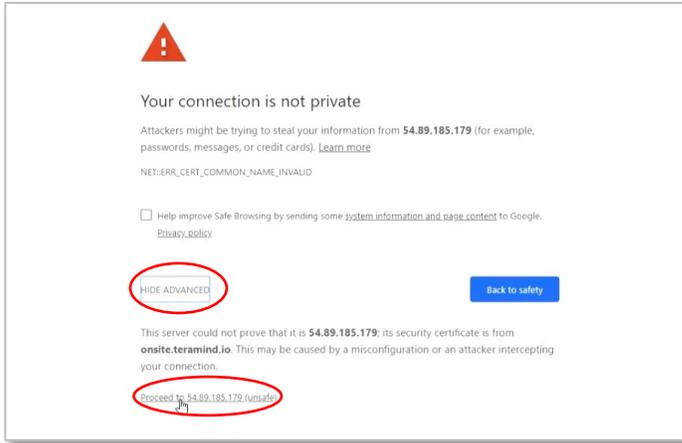
Step 6-4

After Teramind has finished the database initializations, it will ask, 'Use S3 for storing data?'. Type **Yes**.

Type **Yes** to the 'Create new buckets?' prompt.

You will be creating **three buckets**: one for screen recordings, one for user content i.e. attachments or printer docs, and the third one for application objects. Give unique names for each bucket.

Finally, you will be presented with a **https link**. Click this link to continue.



Step 6-5

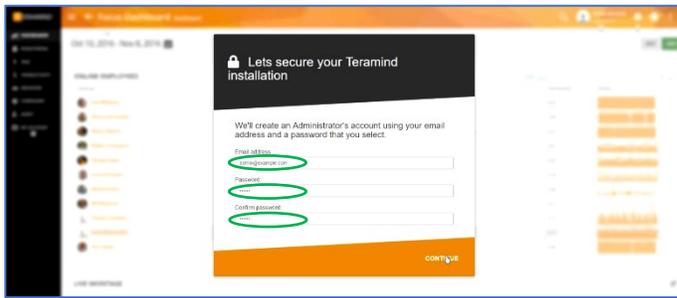
When you click the Teramind server link from the previous step, a browser window will open. You may be displayed a warning message. This is because you are using a HTTPS connection without a SSL certificate. Most browsers will allow you continue to with an override action.

If you are using Google Chrome, click the **ADVANCED** link on the page and select the **Proceed to** option.



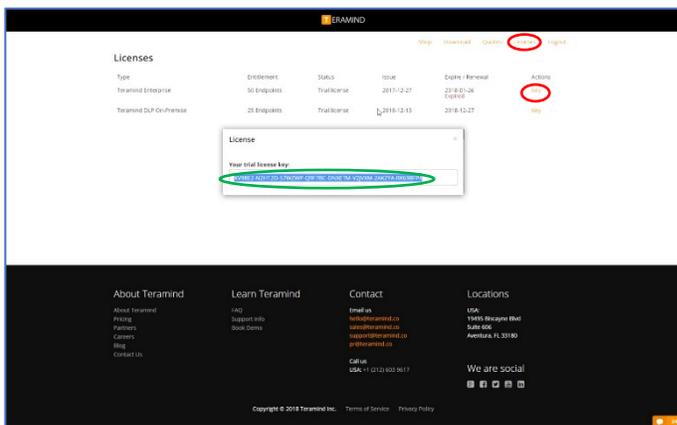
Step 6-6

On the 'Welcome to Teramind' screen, select your **language** and **timezone** and click **CONTINUE**.



Step 6-7

On the 'Let's secure your Teramind installation' screen, enter an **email** and a **password** to be used as your Admin account.



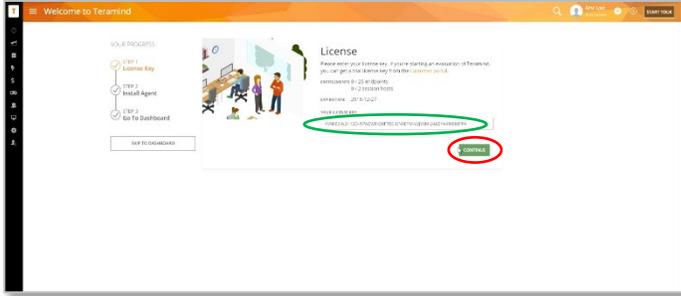
Step 6-8

Open a new browser tab and go to: <https://www.teramind.co/portal>. Login with the admin email and password.

Click the **Licenses** tab.

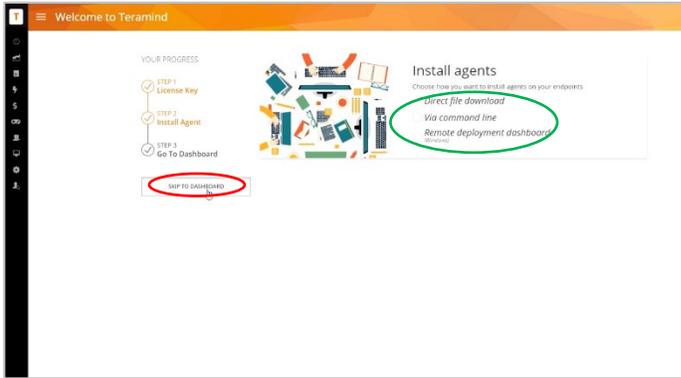
From the list of licenses, click the **Key** link under the 'Actions' column. A pop-up will display the license key.

Copy the **license key** or write it down.



Step 6-9

Go back to your Teramind Dashboard. Enter the **license key** and click the **CONTINUE** button.



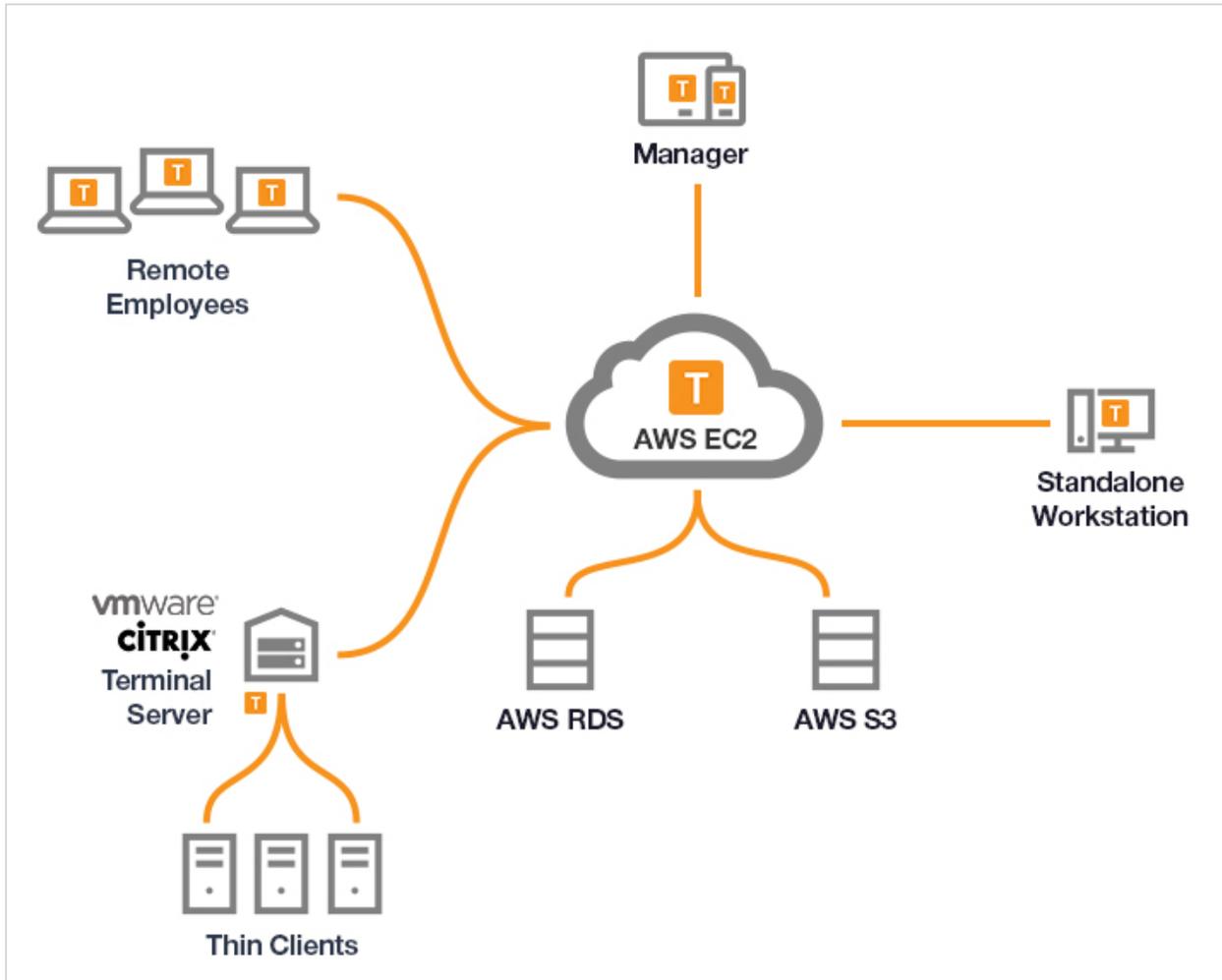
Step 6-10

At this stage, you can install the Teramind agent and start monitoring the targeted user(s). Or, you can do it later.

To install the agent, click one of the **options** under 'Install agents'. If you need help installing the agent, follow the instructions under the '[Teramind Agent](#)' section on Teramind's User Guide at: <https://www.teramind.co/user-guide>. Or, watch the tutorial video titled, '[Downloading and Installing Teramind's Hidden Agent](#)' on the Teramind YouTube channel.

To install the agent at a later time, click the **SKIP TO DASHBOARD** button.

Architecture



Technical Specifications, Licensing and Support

Data centers	<p>Teramind on AWS is available on the following datacenters/regions. We recommend you choose a region closest to you for faster service and lower latency:</p> <ul style="list-style-type: none">• US East (N. Virginia)• US East (Ohio)• US West (N. California)• US West (Oregon)• Canada (Central)• EU (Frankfurt)• EU (Ireland)• EU (London)• EU (Paris)• Asia Pacific (Singapore)• Asia Pacific (Sydney)• Asia Pacific (Seoul)• Asia Pacific (Tokyo)• Asia Pacific (Mumbai)• South America (Sao Paulo)
OS	64-bit Linux/Unix, Debian 9.5 AMI.
Databases	EBS volume by default. Optionally S3, RDS/Postgres can be used as screen/audio recording storage.
Instances	For a typical deployment Teramind recommends m4.xlarge for \$0.20 EC2/hr Various other EC2 instances types (t2, t3, m3 etc.) for difference sizes (micro, large, extra large etc.) are also available.
Teramind license	BYOL (bring your own Teramind license). Click here to try or subscribe to Teramind on-prem/private-cloud.
Support	Basic infrastructure support is available for free from Amazon. Teramind support is included as part of the Teramind subscription.



If you encounter any issues deploying Teramind on AWS, please contact: support@termind.co.