

Icinga 2, Icinga Web 2 and Director Kickstart on CentOS 7

User admin Pass : admin

```
[root@localhost ~]# yum update -y && yum upgrade -y
```

```
[root@localhost ~]#
[root@localhost ~]# yum update -y && yum upgrade -y
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: bd.mirror.vanehost.com
 * epel: epel.mirror.angkasa.id
 * extras: centos.excellmedia.net
 * remi-safe: fr2.rpmfind.net
 * updates: bd.mirror.vanehost.com
base | 3.6 kB | 00:00:00
extras | 2.9 kB | 00:00:00
mariadb | 3.4 kB | 00:00:00
remi-safe | 3.0 kB | 00:00:00
updates | 2.9 kB | 00:00:00
(1/4): mariadb/updateinfo | 5.9 kB | 00:00:01
(2/4): mariadb/primary_db | 65 kB | 00:00:01
(3/4): updates/7/x86_64/primary_db | 19 MB | 00:00:03
(4/4): remi-safe/primary_db | 2.3 MB | 00:00:06
Resolving Dependencies
```

```
[root@localhost ~]# yum install -y https://packages.icinga.com/epel/icinga-rpm-release-7-latest.noarch.rpm
```

```
[root@localhost ~]#
[root@localhost ~]# yum install -y https://packages.icinga.com/epel/icinga-rpm-release-7-latest.noarch.rpm
Loaded plugins: fastestmirror, langpacks
icinga-rpm-release-7-latest.noarch.rpm | 5.1 kB | 00:00:00
Examining /var/tmp/yum-root-atMjPd/icinga-rpm-release-7-latest.noarch.rpm: icinga-rpm-release-7-4.el7.icinga.noarch
Marking /var/tmp/yum-root-atMjPd/icinga-rpm-release-7-latest.noarch.rpm to be installed
Resolving Dependencies
--> Running transaction check
--> Package icinga-rpm-release.noarch 0:7-4.el7.icinga will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package Arch Version Repository Size
=====
Installing:
icinga-rpm-release noarch 7-4.el7.icinga /icinga-rpm-release-7-latest.noarch 2.6 k
=====
Transaction Summary
=====
Install 1 Package
```

```
[root@localhost ~]# yum install -y epel-release
```

```
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# yum install -y epel-release
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: bd.mirror.vanehost.com
 * epel: epel.mirror.angkasa.id
 * extras: centos.excellmedia.net
 * remi-safe: fr2.rpmfind.net
 * updates: bd.mirror.vanehost.com
Package epel-release-7-14.noarch already installed and latest version
Nothing to do
```

```
[root@localhost ~]# yum install -y git curl make gcc wget nano vim net-tools tar unzip zip python-devel
python-pip python-setuptools
```

```
[root@localhost ~]#
[root@localhost ~]# yum install -y git curl make gcc wget nano vim net-tools tar unzip zip python-devel python-pip python-setuptools
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: bd.mirror.vanehost.com
 * epel: epel.mirror.angkasa.id
 * extras: centos.excellmedia.net
 * remi-safe: fr2.rpmfind.net
 * updates: bd.mirror.vanehost.com
Package curl-7.29.0-59.el7_9.1.x86_64 already installed and latest version
Package 1:make-3.82-24.el7.x86_64 already installed and latest version
Package wget-1.14-18.el7_6.1.x86_64 already installed and latest version
Package nano-2.3.1-10.el7.x86_64 already installed and latest version
Package 2:vim-enhanced-7.4.629-8.el7_9.x86_64 already installed and latest version
Package net-tools-2.0-0.25.20131004git.el7.x86_64 already installed and latest version
Package 2:tar-1.26-35.el7.x86_64 already installed and latest version
Package unzip-6.0-24.el7_9.x86_64 already installed and latest version
Package zip-3.0-11.el7.x86_64 already installed and latest version
Package python-setuptools-0.9.8-7.el7.noarch already installed and latest version
Resolving Dependencies
```

[root@localhost ~]# yum install -y icinga2

```
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# yum install -y icinga2
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: bd.mirror.vanehost.com
 * epel: epel.mirror.angkasa.id
 * extras: centos.excellmedia.net
 * remi-safe: fr2.rpmfind.net
 * updates: bd.mirror.vanehost.com
Resolving Dependencies
--> Running transaction check
--> Package icinga2.x86_64 0:2.13.6-1.el7 will be installed
--> Processing Dependency: icinga2-common = 2.13.6-1.el7 for package: icinga2-2.13.6-1.el7.x86_64
--> Processing Dependency: icinga2-bin = 2.13.6-1.el7 for package: icinga2-2.13.6-1.el7.x86_64
--> Running transaction check
```

[root@localhost ~]# systemctl enable icinga2

Created symlink from /etc/systemd/system/multi-user.target.wants/icinga2.service to /usr/lib/systemd/system/icinga2.service.

[root@localhost ~]# systemctl start icinga2

[root@localhost ~]# systemctl status icinga2

```
Complete!
[root@localhost ~]# systemctl enable icinga2
Created symlink from /etc/systemd/system/multi-user.target.wants/icinga2.service to /usr/lib/systemd/system/icinga2.service.
[root@localhost ~]# systemctl start icinga2
[root@localhost ~]# systemctl status icinga2
● icinga2.service - Icinga host/service/network monitoring system
   Loaded: loaded (/usr/lib/systemd/system/icinga2.service; enabled; vendor preset: disabled)
   Active: active (running) since Thu 2022-12-22 12:48:15 +06; 2s ago
     Process: 124936 ExecStartPre=/usr/lib/icinga2/prepare-dirs /etc/sysconfig/icinga2 (code=exited, status=0/SUCCESS)
    Main PID: 124945 (icinga2)
      Status: "Startup finished."
        Tasks: 13
       CGroup: /system.slice/icinga2.service
               └─124945 /usr/lib64/icinga2/sbin/icinga2 --no-stack-rlimit daemon --close-stdio -e /var/log/icinga2/error.log
                 └─124972 /usr/lib64/icinga2/sbin/icinga2 --no-stack-rlimit daemon --close-stdio -e /var/log/icinga2/error.log
                   └─124977 /usr/lib64/icinga2/sbin/icinga2 --no-stack-rlimit daemon --close-stdio -e /var/log/icinga2/error.log
```

```
[root@localhost ~]# yum install -y nagios-plugins-all
```

```
[root@localhost ~]#  
[root@localhost ~]# yum install -y nagios-plugins-all  
Loaded plugins: fastestmirror, langpacks  
Loading mirror speeds from cached hostfile  
* base: bd.mirror.vanehost.com  
* epel: epel.mirror.angkasa.id  
* extras: centos.excellmedia.net  
* remi-safe: fr2.rpmfind.net  
* updates: bd.mirror.vanehost.com  
Resolving Dependencies  
--> Running transaction check  
--> Package nagios-plugins-all.x86_64 0:2.3.3-2.el7 will be installed  
--> Processing Dependency: nagios-plugins-breeze for package: nagios-plugins-all-2.3.3-2.el7.x86_64  
--> Processing Dependency: nagios-plugins-by_ssh for package: nagios-plugins-all-2.3.3-2.el7.x86_64  
--> Processing Dependency: nagios-plugins-dhcp for package: nagios-plugins-all-2.3.3-2.el7.x86_64  
--> Processing Dependency: nagios-plugins-dig for package: nagios-plugins-all-2.3.3-2.el7.x86_64  
--> Processing Dependency: nagios-plugins-disk for package: nagios-plugins-all-2.3.3-2.el7.x86_64
```

```
[root@localhost ~]# icinga2 daemon -C
```

```
[root@localhost ~]# icinga2 daemon -C  
[2022-12-22 12:48:56 +0600] information/cli: Icinga application loader (version: r2.13.6-1)  
[2022-12-22 12:48:56 +0600] information/cli: Loading configuration file(s).  
[2022-12-22 12:48:56 +0600] information/ConfigItem: Committing config item(s).  
[2022-12-22 12:48:56 +0600] information/ConfigItem: Instantiated 1 NotificationComponent.  
[2022-12-22 12:48:56 +0600] information/ConfigItem: Instantiated 1 CheckerComponent.  
[2022-12-22 12:48:56 +0600] information/ConfigItem: Instantiated 1 UserGroup.  
[2022-12-22 12:48:56 +0600] information/ConfigItem: Instantiated 3 TimePeriods.  
[2022-12-22 12:48:56 +0600] information/ConfigItem: Instantiated 1 User.  
[2022-12-22 12:48:56 +0600] information/ConfigItem: Instantiated 11 Services.  
[2022-12-22 12:48:56 +0600] information/ConfigItem: Instantiated 3 ServiceGroups.  
[2022-12-22 12:48:56 +0600] information/ConfigItem: Instantiated 1 ScheduledDowntime.  
[2022-12-22 12:48:56 +0600] information/ConfigItem: Instantiated 3 Zones.  
[2022-12-22 12:48:56 +0600] information/ConfigItem: Instantiated 2 NotificationCommands.  
[2022-12-22 12:48:56 +0600] information/ConfigItem: Instantiated 12 Notifications.  
[2022-12-22 12:48:56 +0600] information/ConfigItem: Instantiated 1 Host.  
[2022-12-22 12:48:56 +0600] information/ConfigItem: Instantiated 1 IcingaApplication.  
[2022-12-22 12:48:56 +0600] information/ConfigItem: Instantiated 2 HostGroups.  
[2022-12-22 12:48:56 +0600] information/ConfigItem: Instantiated 1 Endpoint.  
[2022-12-22 12:48:56 +0600] information/ConfigItem: Instantiated 1 FileLogger.  
[2022-12-22 12:48:56 +0600] information/ConfigItem: Instantiated 244 CheckCommands.  
[2022-12-22 12:48:56 +0600] information/ScriptGlobal: Dumping variables to file '/var/cache/icinga2/icinga2.vars'  
[2022-12-22 12:48:56 +0600] information/cli: Finished validating the configuration file(s).  
[root@localhost ~]#
```

```
[root@localhost ~]# yum install -y icinga2-selinux
```

```
[root@localhost ~]#  
[root@localhost ~]# yum install -y icinga2-selinux  
Loaded plugins: fastestmirror, langpacks  
Loading mirror speeds from cached hostfile  
* base: bd.mirror.vanehost.com  
* epel: epel.mirror.angkasa.id  
* extras: centos.excellmedia.net  
* remi-safe: fr2.rpmfind.net  
* updates: bd.mirror.vanehost.com  
Resolving Dependencies  
--> Running transaction check  
--> Package icinga2-selinux.x86_64 0:2.13.6-1.el7 will be installed  
--> Processing Dependency: icinga-selinux-common for package: icinga2-selinux-2.13.6-1.el7.x86_64  
--> Running transaction check  
--> Package icinga-selinux-common.noarch 0:1.0.0-1.el7.icinga will be installed  
--> Finished Dependency Resolution  
  
Dependencies Resolved  
  
=====
```

Package	Arch	Version	Repository	Size
Installing: icinga2-selinux	x86_64	2.13.6-1.el7	icinga-stable-release	55 k
Installing for dependencies: icinga-selinux-common	noarch	1.0.0-1.el7.icinga	icinga-stable-release	2.3 k

```
=====
```

Transaction Summary

```
=====
```

Install 1 Package (+1 Dependent package)

```
[root@localhost ~]# firewall-cmd --add-service=http && firewall-cmd --permanent --add-service=http
Success
Success
```

```
[root@localhost ~]# yum install -y vim-icinga2
```

```
[root@localhost ~]# yum install -y vim-icinga2
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: bd.mirror.vanehost.com
 * epel: epel.mirror.angkasa.id
 * extras: centos.excellmedia.net
 * remi-safe: fr2.rpmfind.net
 * updates: bd.mirror.vanehost.com
Resolving Dependencies
--> Running transaction check
---> Package vim-icinga2.x86_64 0:2.13.6-1.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package                               Arch              Version            Repository          Size
=====
Installing:
vim-icinga2                           x86_64            2.13.6-1.el7       icinga-stable-release 7.1 k
=====
Transaction Summary
=====
Install 1 Package
```

```
[root@localhost ~]# vim /etc/icinga2/conf.d/templates.conf
```

```
[root@localhost ~]# yum install -y mariadb-server mariadb
```

```
[root@localhost ~]# systemctl enable mariadb
```

```
[root@localhost ~]# systemctl start mariadb
```

```
[root@localhost ~]# systemctl status mariadb
```

```
[root@localhost ~]#
[root@localhost ~]# systemctl status mariadb
● mariadb.service - mariadb 10.4.27 database server
   Loaded: loaded (/usr/lib/systemd/system/mariadb.service; enabled; vendor preset: disabled)
   Drop-In: /etc/systemd/system/mariadb.service.d
            └─migrated-from-my.cnf-settings.conf
   Active: active (running) since Thu 2022-12-22 12:44:27 +06; 8min ago
     Docs: man:mysqld(8)
           https://mariadb.com/kb/en/library/systemd/
  Main PID: 99002 (mysqld)
    Status: "Taking your SQL requests now..."
   CGroup: /system.slice/mariadb.service
           └─99002 /usr/sbin/mysqld
```

```
[root@localhost ~]# mysql_secure_installation
```

```
[root@localhost ~]# mysql -u root -p
```

Enter password:

```
[root@localhost ~]# yum install icinga2-ido-mysql
```

```
[root@localhost ~]#
[root@localhost ~]# yum install icinga2-ido-mysql
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: bd.mirror.vanehost.com
 * epel: epel.mirror.angkasa.id
 * extras: centos.excellmedia.net
 * remi-safe: fr2.rpmfind.net
 * updates: bd.mirror.vanehost.com
Resolving Dependencies
--> Running transaction check
--> Package icinga2-ido-mysql.x86_64 0:2.13.6-1.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package                               Arch              Version            Repository          Size
=====
Installing:
icinga2-ido-mysql                     x86_64            2.13.6-1.el7       icinga-stable-release 120 k
=====
Transaction Summary
=====
Install 1 Package
=====
```

```
[root@localhost ~]# mysql -u root -p
```

Enter password:

```
MariaDB [(none)]> CREATE DATABASE icinga;
```

Query OK, 1 row affected (0.015 sec)

```
MariaDB [(none)]> GRANT SELECT, INSERT, UPDATE, DELETE, DROP, CREATE VIEW, INDEX, EXECUTE ON
icinga.* TO 'icinga'@'localhost' IDENTIFIED BY 'icinga';
```

Query OK, 0 rows affected (0.016 sec)

```
MariaDB [(none)]> CREATE DATABASE icingaweb;
```

Query OK, 1 row affected (0.001 sec)

```
MariaDB [(none)]> GRANT ALL PRIVILEGES ON icingaweb.* TO 'icingaweb'@'localhost' IDENTIFIED BY
'icingaweb';
```

Query OK, 0 rows affected (0.001 sec)

```
MariaDB [(none)]> FLUSH PRIVILEGES;
```

Query OK, 0 rows affected (0.003 sec)

```
MariaDB [(none)]> exit
```

```

[root@localhost ~]# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 23
Server version: 10.4.27-MariaDB MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
MariaDB [(none)]>
MariaDB [(none)]> CREATE DATABASE icinga;
Query OK, 1 row affected (0.015 sec)

MariaDB [(none)]>
MariaDB [(none)]>
MariaDB [(none)]> GRANT SELECT, INSERT, UPDATE, DELETE, DROP, CREATE VIEW, INDEX, EXECUTE ON icinga.* TO 'icinga'@'localhost' IDENTIFIED BY 'icinga';
Query OK, 0 rows affected (0.016 sec)

MariaDB [(none)]>
MariaDB [(none)]>
MariaDB [(none)]>
MariaDB [(none)]> CREATE DATABASE icingaweb;
Query OK, 1 row affected (0.001 sec)

MariaDB [(none)]>
MariaDB [(none)]>
MariaDB [(none)]> GRANT ALL PRIVILEGES ON icingaweb.* TO 'icingaweb'@'localhost' IDENTIFIED BY 'icingaweb';
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]>
MariaDB [(none)]>
MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.005 sec)

MariaDB [(none)]>
MariaDB [(none)]>
MariaDB [(none)]> exit
Bye

```

[root@localhost ~]# mysql -u root -p icinga < /usr/share/icinga2-ido-mysql/schema/mysql.sql

Enter password:

[root@localhost ~]# icinga2 feature enable ido-mysql

Enabling feature ido-mysql. Make sure to restart Icinga 2 for these changes to take effect.

[root@localhost ~]# systemctl restart icinga2

[root@localhost ~]# yum install -y httpd

[root@localhost ~]# systemctl enable httpd

[root@localhost ~]# systemctl start httpd

[root@localhost ~]# icinga2 api setup

```

[root@localhost ~]#
[root@localhost ~]# icinga2 api setup
information/cli: Generating new CA.
information/base: Writing private key to '/var/lib/icinga2/ca//ca.key'.
information/base: Writing X509 certificate to '/var/lib/icinga2/ca//ca.crt'.
information/cli: Generating new CSR in '/var/lib/icinga2/certs//localhost.localdomain.csr'.
information/base: Writing private key to '/var/lib/icinga2/certs//localhost.localdomain.key'.
information/base: Writing certificate signing request to '/var/lib/icinga2/certs//localhost.localdomain.csr'.
information/cli: Signing CSR with CA and writing certificate to '/var/lib/icinga2/certs//localhost.localdomain.crt'.
information/pki: Writing certificate to file '/var/lib/icinga2/certs//localhost.localdomain.crt'.
information/cli: Copying CA certificate to '/var/lib/icinga2/certs//ca.crt'.
information/cli: Adding new ApiUser 'root' in '/etc/icinga2/conf.d/api-users.conf'.
information/cli: Reading '/etc/icinga2/icinga2.conf'.
information/cli: Enabling the 'api' feature.
Enabling feature api. Make sure to restart Icinga 2 for these changes to take effect.
information/cli: Updating 'NodeName' constant in '/etc/icinga2/constants.conf'.
information/cli: Created backup file '/etc/icinga2/constants.conf.orig'.
information/cli: Updating 'ZoneName' constant in '/etc/icinga2/constants.conf'.
information/cli: Backup file '/etc/icinga2/constants.conf.orig' already exists. Skipping backup.
Done.

Now restart your Icinga 2 daemon to finish the installation!

```

```
[root@localhost ~]# vim /etc/icinga2/conf.d/api-users.conf
```

```
object ApiUser "icingaweb2" {  
    password = "Wijsn8Z9eRs5E25d"  
    permissions = [ "status/query", "actions/*", "objects/modify/*", "objects/query/*" ]  
}
```

```
/**  
 * The ApiUser objects are used for authentication against the API.  
 */  
object ApiUser "icingaweb2" {  
    password = "Wijsn8Z9eRs5E25d"  
    permissions = [ "status/query", "actions/*", "objects/modify/*", "objects/query/*" ]  
}
```

```
[root@localhost ~]# vim /etc/icinga2/features-enabled/ido-mysql.conf
```

```
/**  
 * The IdoMysqlConnection type implements MySQL support  
 * for DB IDO.  
 */  
object IdoMysqlConnection "ido-mysql" {  
    user = "icinga"  
    password = "icinga"  
    host = "localhost"  
    database = "icinga"  
}
```

```
/**  
 * The IdoMysqlConnection type implements MySQL support  
 * for DB IDO.  
 */  
object IdoMysqlConnection "ido-mysql" {  
    user = "icinga"  
    password = "icinga"  
    host = "localhost"  
    database = "icinga"  
}
```

```
[root@localhost ~]# systemctl restart icinga2
```

```
[root@localhost ~]# yum install -y centos-release-scl
```

```
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# yum install -y centos-release-scl
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: bd.mirror.vanehost.com
 * bases: bd.mirror.vanehost.com
 * epel: epel.mirror.angkasa.id
 * extras: centos.excellmedia.net
 * remi-safe: fr2.rpmfind.net
 * updates: bd.mirror.vanehost.com
Resolving Dependencies
--> Running transaction check
--> Package centos-release-scl.noarch 0:2-3.el7.centos will be installed
--> Processing Dependency: centos-release-scl-rh for package: centos-release-scl-2-3.el7.centos.noarch
--> Running transaction check
--> Package centos-release-scl-rh.noarch 0:2-3.el7.centos will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package                                Arch                                Version                                Repository                                Size
=====
Installing:
centos-release-scl                    noarch                              2-3.el7.centos                        extras                                    12 k
Installing for dependencies:
centos-release-scl-rh                 noarch                              2-3.el7.centos                        extras                                    12 k
=====
Transaction Summary
=====
Install 1 Package (+1 Dependent package)
```

```
[root@localhost ~]# yum install -y icingaweb2 icingacl
```

```
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# yum install -y icingaweb2 icingacl
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: bd.mirror.vanehost.com
 * centos-scl-rh: mirrors.glu.edu.cn
 * centos-scl-scl: bd.mirror.vanehost.com
 * epel: epel.mirror.angkasa.id
 * extras: centos.excellmedia.net
 * remi-safe: fr2.rpmfind.net
 * updates: bd.mirror.vanehost.com
centos-scl-rh                                | 3.0 kB  00:00:00
centos-scl-scl                                | 3.0 kB  00:00:00
(1/2): centos-scl-scl/x86_64/primary.db       | 300 kB  00:00:00
(2/2): centos-scl-rh/x86_64/primary.db       | 3.4 MB  00:00:00
Resolving Dependencies
--> Running transaction check
--> Package icingacl.noarch 0:2.11.3-2.el7.icinga will be installed
--> Processing Dependency: icingaweb2-common = 2.11.3-2.el7.icinga for package: icingacl-2.11.3-2.el7.icinga.noarch
--> Processing Dependency: php-icinga = 2.11.3-2.el7.icinga for package: icingacl-2.11.3-2.el7.icinga.noarch
--> Processing Dependency: rh-php73-php-cli >= 7.2 for package: icingacl-2.11.3-2.el7.icinga.noarch
--> Processing Dependency: icinga-php-library >= 0.9.0-1 for package: icingacl-2.11.3-2.el7.icinga.noarch
--> Processing Dependency: icinga-php-thirdparty >= 0.11.0-1 for package: icingacl-2.11.3-2.el7.icinga.noarch
--> Processing Dependency: icinga-l10n >= 1.1.0-1 for package: icingacl-2.11.3-2.el7.icinga.noarch
--> Processing Dependency: /opt/rh/rh-php73/root/usr/bin/php for package: icingacl-2.11.3-2.el7.icinga.noarch
--> Package icingaweb2.noarch 0:2.11.3-2.el7.icinga will be installed
```

```
[root@localhost ~]# yum install -y icingaweb2-selinux
```

```
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# yum install -y icingaweb2-selinux
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: bd.mirror.vanehost.com
 * centos-scl-rh: mirrors.glu.edu.cn
 * centos-scl-scl: bd.mirror.vanehost.com
 * epel: epel.mirror.angkasa.id
 * extras: centos.excellmedia.net
 * remi-safe: fr2.rpmfind.net
 * updates: bd.mirror.vanehost.com
Resolving Dependencies
--> Running transaction check
--> Package icingaweb2-selinux.noarch 0:2.11.3-2.el7.icinga will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package                                Arch                                Version                                Repository                                Size
=====
Installing:
icingaweb2-selinux                    noarch                              2.11.3-2.el7.icinga                  icinga-stable-release                    18 k
=====
Transaction Summary
=====
Install 1 Package
Total download size: 18 k
```



```
[root@localhost ~]# yum install -y rh-php73-php-mysqlnd rh-php73-php-fpm sclo-php73-php-pecl-imagick rh-php73-php-ldap rh-php73-php-pgsql rh-php73-php-xmlrpc rh-php73-php-intl rh-php73-php-gd rh-php73-php-pdo rh-php73-php-soap rh-php73-php-posix rh-php73-php-cli
```

```
[root@localhost ~]# yum install -y rh-php73-php-mysqlnd rh-php73-php-fpm sclo-php73-php-pecl-imagick rh-php73-php-ldap rh-php73-php-pgsql rh-php73-php-xmlrpc rh-php73-php-intl rh-php73-php-gd rh-php73-php-pdo rh-php73-php-soap rh-php73-php-posix rh-php73-php-cli
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: bd.mirror.vanehost.com
 * centos-sclo-rh: mirrors.glu.edu.cn
 * centos-sclo-sclo: bd.mirror.vanehost.com
 * epel: epel.mirror.angkasa.id
 * extras: centos.excellmedia.net
 * remi-safe: fr2.rpmfind.net
 * updates: bd.mirror.vanehost.com
Package rh-php73-php-mysqlnd-7.3.33-1.el7.x86_64 already installed and latest version
Package rh-php73-php-fpm-7.3.33-1.el7.x86_64 already installed and latest version
Package rh-php73-php-ldap-7.3.33-1.el7.x86_64 already installed and latest version
Package rh-php73-php-pgsql-7.3.33-1.el7.x86_64 already installed and latest version
Package rh-php73-php-intl-7.3.33-1.el7.x86_64 already installed and latest version
Package rh-php73-php-gd-7.3.33-1.el7.x86_64 already installed and latest version
Package rh-php73-php-pdo-7.3.33-1.el7.x86_64 already installed and latest version
Package rh-php73-php-soap-7.3.33-1.el7.x86_64 already installed and latest version
Package rh-php73-php-cli-7.3.33-1.el7.x86_64 already installed and latest version
Resolving Dependencies
```

```
[root@localhost ~]# systemctl start rh-php73-php-fpm.service
```

```
[root@localhost ~]# systemctl enable rh-php73-php-fpm.service
```

Created symlink from /etc/systemd/system/multi-user.target.wants/rh-php73-php-fpm.service to /usr/lib/systemd/system/rh-php73-php-fpm.service.

```
[root@localhost ~]# systemctl restart httpd
```

```
[root@localhost ~]# systemctl restart rh-php73-php-fpm.service
```

```
[root@localhost ~]# icingacli setup token create
```

The newly generated setup token is: 5ac97fe5cffb0025

```
[root@localhost ~]#
[root@localhost ~]# icingacli setup token create
The newly generated setup token is: 5ac97fe5cffb0025
[root@localhost ~]#
[root@localhost ~]#
```

Install Source link:

<https://icinga.com/blog/2020/07/10/icinga-2-icinga-web-2-and-director-kickstart-on-centos-7/>

Icinga 2, Icinga Web 2 and Director Kickstart on CentOS 7



INTRODUCTION

The easiest way to get started with Icinga is a single-node installation. If you are new to the Icinga world, here you have a kickstart for installing Icinga 2, Icinga Web 2 and Icinga Director on CentOS 7. With these steps you will have a ready Icinga environment for monitoring your infrastructure.

Step 1: Now update / upgrade your CentOS Linux

```
yum update -y && yum upgrade -y
```

Step 2: Install Icinga repository

```
yum install -y https://packages.icinga.com/epel/icinga-rpm-release-7-latest.noarch.rpm
```

Step 3: Install EPEL repository

```
yum install -y epel-release
```

Step 4: Let's install some tools we could need later in our system

```
yum install -y git curl make gcc wget nano vim net-tools tar unzip zip python-devel python-pip python-setuptools
```

Step 5: Install Icinga 2, enable and start the Icinga 2 service

```
yum install -y icinga2  
systemctl enable icinga2  
systemctl start icinga2  
systemctl status icinga2
```

Step 6: Install plugins-all

```
yum install -y nagios-plugins-all
```

Step 7: When you make a change into Icinga files and you want to confirm all is correct before restarting, you can run the following command

```
icinga2 daemon -C
```

Step 8: If you are going to use SELINUX with Icinga 2, you need to install the following packages, and set rules for port 80 and 443

```
yum install -y icinga2-selinux
```

Now proceed to apply firewall rules for port 80, as a best practice you should be using https and open 443 as well.

```
firewall-cmd --add-service=http && firewall-cmd --permanent --add-service=http
```

Step 9: Let's configure vim to use Icinga syntax colors

```
yum install -y vim-icinga2
```

Now, create the following file `vim ~/.vimrc` and add the following: *syntax on*
Now let's test that syntax colors are working, open the file *templates.conf* with vim editor:

```
vim /etc/icinga2/conf.d/templates.conf
```

Step 10: Installing and configuring MySQL as database for our Icinga:

```
yum install -y mariadb-server mariadb
systemctl enable mariadb
systemctl start mariadb
mysql_secure_installation
```

```
[ below details of this step... ]
Enter current password for root (enter for none): [ Hit enter ]
Set root password? [Y/n] Y
New password: [ Enter new password ]
Re-enter new password: [ Enter again new password ]
Remove anonymous users? [Y/n] Y
Disallow root login remotely? [Y/n] Y
Remove test database and access to it? [Y/n] Y
Reload privilege tables now? [Y/n] Y
```

Now let's install IDO for MySQL

```
yum install icinga2-id0-mysql
```

Login to your MySQL

```
mysql -u root -p
```

```
CREATE DATABASE icinga;
```

Now, configure the permissions for the database created in the step before:

```
GRANT SELECT, INSERT, UPDATE, DELETE, DROP, CREATE VIEW, INDEX,
EXECUTE ON icinga.* TO 'icinga'@'localhost' IDENTIFIED BY 'icinga';
```

Create the database for Icinga Web 2:

```
CREATE DATABASE icingaweb;
```

Now, configure permissions:

```
GRANT ALL PRIVILEGES ON icingaweb.* TO 'icingaweb'@'localhost' IDENTIFIED BY
'icingaweb';
```

Deploy privileges:

```
FLUSH PRIVILEGES;
```

Now, quit from the database:

QUIT

Step 11: Now we need to import the Icinga 2 schema for our MySQL.

```
mysql -u root -p icinga < /usr/share/icinga2-ido-mysql/schema/mysql.sql
```

Step 12: Enable the ido-mysql module in Icinga

```
icinga2 feature enable ido-mysql
```

Now, restart icinga2 service

```
systemctl restart icinga2
```

Step 13: Install Web Server

```
yum install -y httpd  
systemctl enable httpd  
systemctl start httpd
```

Step 14: Configure Icinga 2 REST ApiUser

```
icinga2 api setup
```

Now, edit the file

```
vim /etc/icinga2/conf.d/api-users.conf
```

and add the following lines:

```
object ApiUser "icingaweb2" {  
    password = "Wijsn8Z9eRs5E25d"  
    permissions = [ "status/query", "actions/*", "objects/modify/*", "objects/query/*" ]  
}
```

Don't forget to edit

```
vim /etc/icinga2/features-enabled/ido-mysql.conf
```

according to the configuration needed. Check an example below:

```
/**
 * The IdoMysqlConnection type implements MySQL support
 * for DB IDO.
 */
object IdoMysqlConnection "ido-mysql" {
    user = "icinga"
    password = "icinga"
    host = "localhost"
    database = "icinga"
}
```

Now proceed to restart icinga2 service

```
systemctl restart icinga2
```

Step 15: Let's install the SCL repository, we'll need it for Icinga Web 2

```
yum install -y centos-release-scl
```

Step 16: Proceed to install Icinga Web and Icinga CLI

```
yum install -y icingaweb2 icingacli
```

Step 17: Install SELINUX for Icinga Web 2 in case you could need it

```
yum install -y icingaweb2-selinux
```

Step 18: Install PHP FPM and other PHP modules we could need

```
yum install -y rh-php73-php-mysqlnd rh-php73-php-fpm sclo-php73-php-pecl-imagick rh-php73-
php-ldap rh-php73-php-pgsql rh-php73-php-xmlrpc rh-php73-php-intl rh-php73-php-gd rh-php73-
php-pdo rh-php73-php-soap rh-php73-php-posix rh-php73-php-cli
```

Then start and enable the service:

```
systemctl start rh-php73-php-fpm.service
systemctl enable rh-php73-php-fpm.service
systemctl restart httpd
systemctl restart rh-php73-php-fpm.service
```

Now, let's create the token for finishing Icinga Web 2 configuration through web interface:

icingacli setup token create

Step 19: Icinga Web 2 Configuration

<http://icinga-server/icingaweb2/>

1. It will request the generated token, paste it and click on next



Welcome to the configuration of Icinga Web 2!

This wizard will guide you through the configuration of Icinga Web 2. Once completed and successfully finished you are able to log in and to explore all the new and stunning features!

Setup Token

Next

Generating a New Setup Token

To run this wizard a user needs to authenticate using a token which is usually provided to him by an administrator who'd followed the instructions below.

In any case, make sure that all of the following applies to your environment:

- A system group called "icingaweb2" exists
- The user "apache" is a member of the system group "icingaweb2"

```
groupadd -r -g icingaweb2;  
usermod -s -G icingaweb2 apache;
```

If you've got the IcingaCLI installed you can do the following:

```
icingacli setup config directory --group icingaweb2;  
icingacli setup token create;
```

In case the IcingaCLI is missing you can create the token manually:

```
su apache -s /bin/bash -c "mkdir -p 2770 /etc/icingaweb2; chgrp icingaweb2 /etc/icingaweb2; head -c 32 /dev/urandom | base64 | tee /etc/icingaweb2/setup.token; chmod 0600 /etc/icingaweb2/setup.token";
```

Please see the Icinga Web 2 documentation for an extensive description on how to access and use this wizard.

2. Monitoring modules is enable by default, you can enable Doc and Translation optionally and then click next



Modules

The following modules were found in your Icinga Web 2 installation. To enable and configure a module, just tick it and click "Next".

Doc	Monitoring	Translation
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Extracts, shows and exports documentation for Icinga Web 2 and its modules.	IDO accessor and UI for your monitoring. This is the initial instalment for a graphical presentation of Icinga environments. The predecessor of Icinga DB.	This module allows developers and translators to translate Icinga Web 2 and its modules for multiple languages. You do not need this module to run an internationalized web frontend. This is only for people who want to contribute translations or translate just their own modules.

Back Next

3. Now all PHP Modules should be green, in case you have some in yellow, it is recommended to fix it before moving forward, if all are green, just click next

icinga

Welcome Modules Requirements Configuration Finish

PHP Module: GD In case you want views being exported to PDF, you'll need the GD extension for PHP. The PHP module GD is available.

PHP Module: Imagick In case you want graphs being exported to PDF as well, you'll need the ImageMagick extension for PHP. The PHP module Imagick is available.

PHP Module: PDO-MySQL To store users or preferences in a MySQL database the PDO-MySQL module for PHP is required. The PHP module PDO-MySQL is available.

Zend database adapter for MySQL The Zend database adapter for MySQL is required to access a MySQL database. The Zend database adapter for MySQL is available.

PHP Module: PDO-PostgreSQL To store users or preferences in a PostgreSQL database the PDO-PostgreSQL module for PHP is required. The PHP module PDO-PostgreSQL is available.

Zend database adapter for PostgreSQL The Zend database adapter for PostgreSQL is required to access a PostgreSQL database. The Zend database adapter for PostgreSQL is available.

Read- and writable configuration directory The Icinga Web 2 configuration directory defaults to "/etc/icingaweb2", if not explicitly set in the environment variable "ICINGAWEB_CONFIGDIR". The directory /etc/icingaweb2 is read- and writable.

Monitoring Module

PHP Module: PDO-MySQL To access the IDO stored in a MySQL database the PDO-MySQL module for PHP is required. The PHP module PDO-MySQL is available.

Zend database adapter for MySQL The Zend database adapter for MySQL is required to access a MySQL database. The Zend database adapter for MySQL is available.

PHP Module: PDO-PostgreSQL To access the IDO stored in a PostgreSQL database the PDO-PostgreSQL module for PHP is required. The PHP module PDO-PostgreSQL is available.

Zend database adapter for PostgreSQL The Zend database adapter for PostgreSQL is required to access a PostgreSQL database. The Zend database adapter for PostgreSQL is available.

PHP Module: cURL To send external commands over Icinga 2's API the cURL module for PHP is required. The PHP module cURL is available.

Back Next Refresh

4. By default we are going to use database authentication

icinga

Welcome Modules Requirements Configuration Finish


Authentication

Please choose how you want to authenticate when accessing Icinga Web 2. Configuring backend specific details follows in a later step.

Authentication Type Database

Back Next

5. Configure the database resource, here we are going to use the credentials we created for icingaweb database. You need to set the parameters localhost, database name, username and password. Before clicking next, you can click on Validate Configuration in order to validate that the credentials are working correctly.



Welcome

Modules

Requirements

Configuration

Finish

Database Resource

i Now please configure the database resource where to store users and user groups.
Note that the database itself does not need to exist at this time as it is going to be created once the wizard is about to be finished.

i The configuration has been successfully validated.

Resource Name *

Database Type *

Host *

Port

Database Name *

Username *


Password *

Character Set

Use SSL ☐

* Required field

6. Let's configure the Authentication Backend, this one was defined in the api-users.conf file, just click next.



Welcome

Modules

Requirements

Configuration

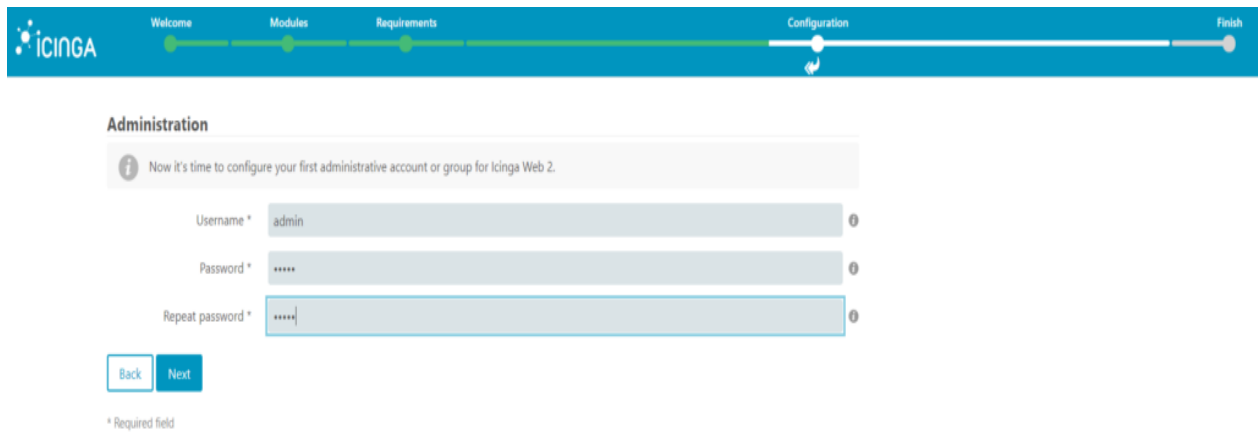
Finish

Authentication Backend

i As you've chosen to use a database for authentication all you need to do now is defining a name for your first authentication backend.

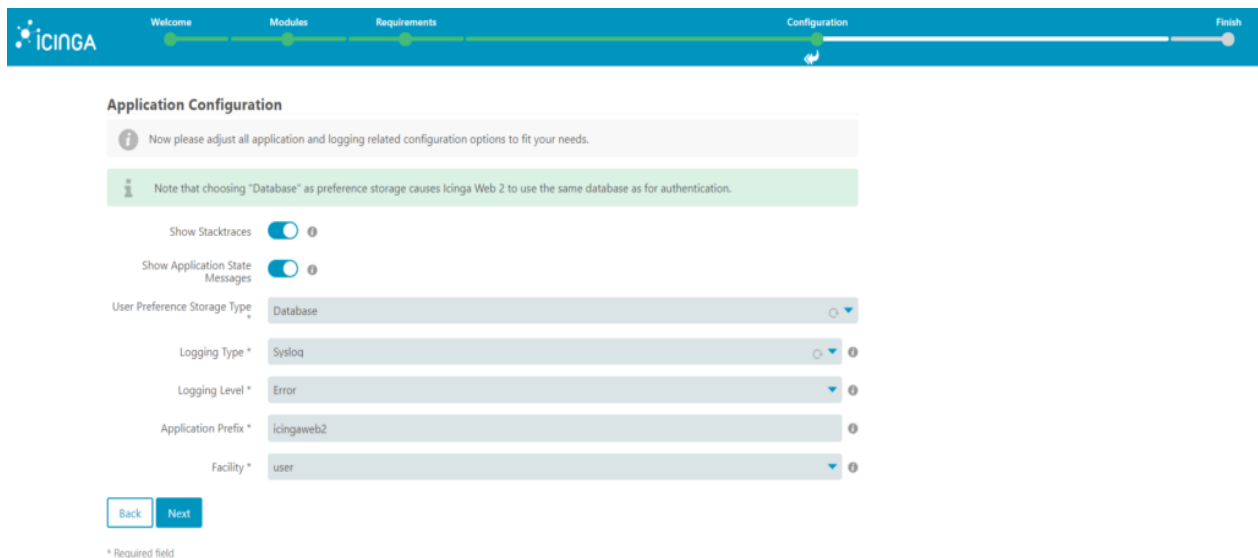
Backend Name

7. In the administration screen you define the username and password for login to the Icinga Web interface.



The screenshot shows the 'Administration' screen in the Icinga Web 2 installation wizard. At the top, a progress bar indicates the current step is 'Configuration', with 'Welcome', 'Modules', 'Requirements', and 'Finish' as previous and subsequent steps. The main heading is 'Administration'. Below it, an information icon and text state: 'Now it's time to configure your first administrative account or group for Icinga Web 2.' The form contains three required fields: 'Username *' with the value 'admin', 'Password *' with masked characters '*****', and 'Repeat password *' also with masked characters '*****'. Each field has a help icon to its right. At the bottom left are 'Back' and 'Next' buttons. A footnote at the bottom left states '* Required field'.

8. In the application configuration screen you just need to click next, you can change adjust it according your needs, but the defaults are fine for now.



The screenshot shows the 'Application Configuration' screen in the Icinga Web 2 installation wizard. The progress bar at the top shows 'Configuration' as the current step. The heading is 'Application Configuration'. An information icon and text state: 'Now please adjust all application and logging related configuration options to fit your needs.' Below this is a green note box: 'Note that choosing "Database" as preference storage causes Icinga Web 2 to use the same database as for authentication.' The configuration options include: 'Show Stacktraces' (toggle on), 'Show Application State Messages' (toggle on), 'User Preference Storage Type *' (dropdown set to 'Database'), 'Logging Type *' (dropdown set to 'Syslog'), 'Logging Level *' (dropdown set to 'Error'), 'Application Prefix *' (text field with 'icingaweb2'), and 'Facility *' (dropdown set to 'user'). Each dropdown or text field has a help icon to its right. At the bottom left are 'Back' and 'Next' buttons. A footnote at the bottom left states '* Required field'.

9. Now you have a resume screen, here just need to click next.



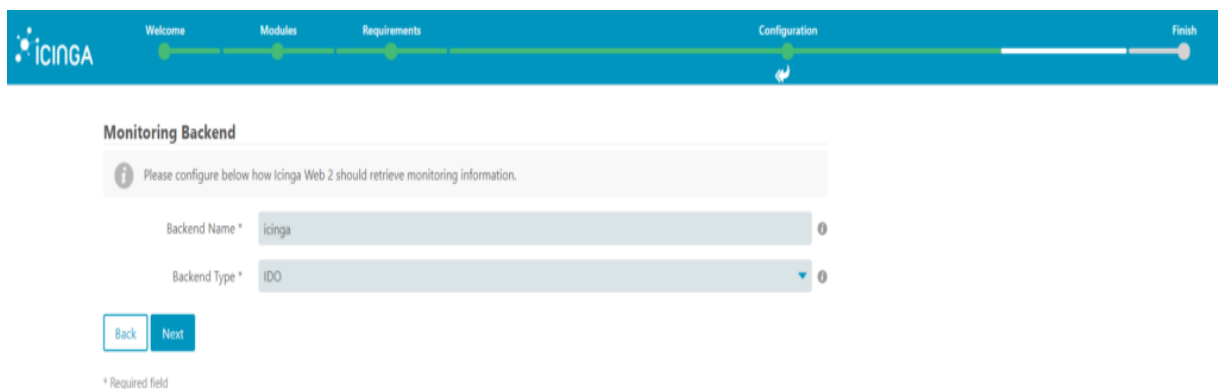
The screenshot shows the Icinga Web 2 Configuration Wizard at the 'Configuration' step. The progress bar at the top indicates the current step. The main content area is divided into three sections: 'Database Setup', 'Application Configuration', and 'Authentication'. 'Database Setup' explains that the 'icingaweb' user will be used to set up the missing schema required by Icinga Web 2 in the 'icingaweb' database. 'Application Configuration' includes 'General' settings (An exception's stacktrace is shown to every user by default, Preferences will be stored using a database) and 'Logging' settings (Type: Syslog, Level: Error, Application Prefix: icingaweb2). 'Authentication' states that users will authenticate using a database, with 'Authentication Backend' set to 'icingaweb2' and 'Administration' rights granted to a new account called 'admin'. Below these sections is a 'Resource' section for the database, showing 'Resource Name: icingaweb_db', 'Database Type: mysql', 'Host: localhost', 'Port: 3306', 'Database Name: icingaweb', 'Username: icingaweb', and 'Password: *****'. At the bottom are 'Back' and 'Next' buttons.

10. Now, you can configure monitoring module for Icinga Web 2, only click next.



The screenshot shows the Icinga Web 2 Configuration Wizard at the 'Configuration' step, specifically the 'Welcome to the configuration of the monitoring module for Icinga Web 2!' screen. The progress bar at the top indicates the current step. The main content area contains a welcome message and a description: 'This is the core module for Icinga Web 2. It offers various status and reporting views with powerful filter capabilities that allow you to keep track of the most important events in your monitoring environment.' At the bottom are 'Back' and 'Next' buttons.

11. Monitoring backend will bring the information by default, here only click next



The screenshot shows the Icinga Web 2 Configuration Wizard at the 'Configuration' step, specifically the 'Monitoring Backend' screen. The progress bar at the top indicates the current step. The main content area contains a message: 'Please configure below how Icinga Web 2 should retrieve monitoring information.' Below this are two required fields: 'Backend Name *' with the value 'icinga' and 'Backend Type *' with the value 'IDO'. At the bottom are 'Back' and 'Next' buttons. A footnote at the bottom left states '* Required field'.

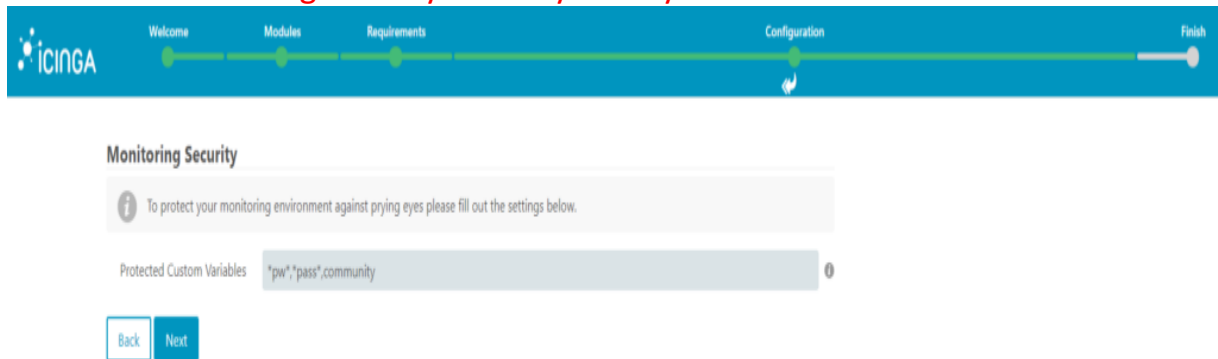
12. Now, let's configure the monitoring IDO Resource, here we'll use the credentials we created for the icinga database. Here you need to set the parameters localhost, database name, username and password. Before clicking next, you can click on Validate Configuration in order to validate that the credentials are working correctly.

The screenshot shows the Icinga 2 web interface with a progress bar at the top indicating the current step is 'Configuration'. The main heading is 'Monitoring IDO Resource'. Below it, there is an information icon and a message: 'Please fill out the connection details below to access the IDO database of your monitoring environment.' A green success message states: 'The configuration has been successfully validated.' A 'Validation Log' section shows the following output: 'Connection to icinga as icinga on localhost: successful', 'have_ssl: DISABLED', 'protocol_version: 10', 'version: 5.5.65-MariaDB', and 'version_compile_os: Linux'. The configuration form includes fields for 'Resource Name' (icinga_ido), 'Database Type' (MySQL), 'Host' (localhost), 'Port' (empty), 'Database Name' (icinga), 'Username' (icinga), 'Password' (masked with dots), and 'Character Set' (empty). A 'Use SSL' toggle is set to 'off'. At the bottom, there are three buttons: 'Back', 'Next', and 'Validate Configuration'.

13. In steps before you defined an api user in the api-user.conf file, now you need to define host, api username and api password. Before clicking next, you can click on Validate Configuration in order to validate that the credentials are working correctly.

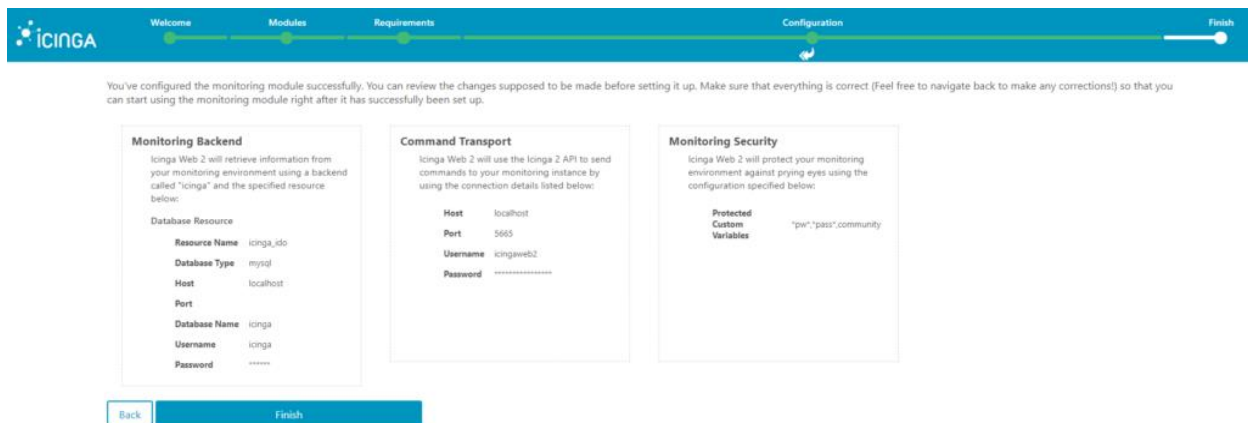
The screenshot shows the Icinga 2 web interface with a progress bar at the top indicating the current step is 'Configuration'. The main heading is 'Command Transport'. Below it, there is an information icon and a message: 'Please define below how you want to send commands to your monitoring instance.' A green success message states: 'The configuration has been successfully validated.' The configuration form includes fields for 'Transport Name' (icinga2), 'Transport Type' (Icinga 2 API), 'Host' (localhost), 'Port' (5665), 'API Username' (icingaweb2), and 'API Password' (masked with dots). At the bottom, there are three buttons: 'Back', 'Next', and 'Validate Configuration'. A footnote at the bottom left states: '* Required field'.

14. In the monitoring security screen you only need to click next.



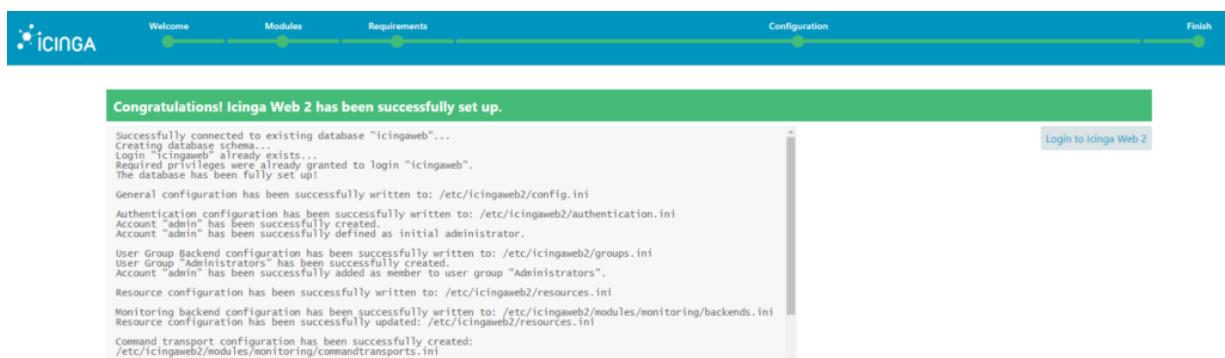
The screenshot shows the 'Monitoring Security' configuration screen in the Icinga Web 2 installer. At the top, a progress bar indicates the current step is 'Configuration'. Below the progress bar, there is a message: 'To protect your monitoring environment against prying eyes please fill out the settings below.' Underneath, a text input field labeled 'Protected Custom Variables' contains the value '*pw","pass",community'. At the bottom left, there are two buttons: 'Back' and 'Next'.

15. Now you have a success screen for Icinga Web 2, you only need to click Finish.



The screenshot shows the 'Configuration' success screen in the Icinga Web 2 installer. At the top, a progress bar indicates the current step is 'Configuration'. Below the progress bar, there is a message: 'You've configured the monitoring module successfully. You can review the changes supposed to be made before setting it up. Make sure that everything is correct (Feel free to navigate back to make any corrections!) so that you can start using the monitoring module right after it has successfully been set up.' Below this message, there are three panels: 'Monitoring Backend', 'Command Transport', and 'Monitoring Security'. Each panel displays the configuration details for that module. At the bottom left, there are two buttons: 'Back' and 'Finish'.

16. You should have now a congratulations screen with a Login to Icinga Web available, only click on that button for logging in into Icinga Web 2.



The screenshot shows the 'Congratulations! Icinga Web 2 has been successfully set up.' screen. At the top, a progress bar indicates the current step is 'Configuration'. Below the progress bar, there is a green banner with the text 'Congratulations! Icinga Web 2 has been successfully set up.' Below the banner, there is a large text area displaying the following information: 'Successfully connected to existing database "icingaweb"... Creating database schema... Login "icingaweb" already exists... Required privileges were already granted to login "icingaweb". The database has been fully set up! General configuration has been successfully written to: /etc/icingaweb2/config.ini Authentication configuration has been successfully written to: /etc/icingaweb2/authentication.ini Account "admin" has been successfully created. Account "admin" has been successfully defined as initial administrator. User Group Backend configuration has been successfully written to: /etc/icingaweb2/groups.ini User Group "Administrators" has been successfully created. Account "admin" has been successfully added as member to user group "Administrators". Resource configuration has been successfully written to: /etc/icingaweb2/resources.ini Monitoring backend configuration has been successfully written to: /etc/icingaweb2/modules/monitoring/backends.ini Resource configuration has been successfully updated: /etc/icingaweb2/resources.ini Command transport configuration has been successfully created: /etc/icingaweb2/modules/monitoring/commandtransports.ini'. At the top right of the text area, there is a button labeled 'Login to Icinga Web 2'.

17. Now you have your login, remember to use the credentials defined in step 7

ICINGA DIRECTOR

```
[root@localhost ~]# mysql -u root -p
```

```
MariaDB [(none)]> CREATE DATABASE director CHARACTER SET 'utf8';
```

```
MariaDB [(none)]> GRANT ALL ON director.* TO 'director'@'localhost' IDENTIFIED BY 'director';
```

```
MariaDB [(none)]> FLUSH PRIVILEGES;
```

```
MariaDB [(none)]> exit
```

```
MariaDB [(none)]>
MariaDB [(none)]> CREATE DATABASE director CHARACTER SET 'utf8';
Query OK, 1 row affected (0.003 sec)

MariaDB [(none)]>
MariaDB [(none)]>
MariaDB [(none)]> GRANT ALL ON director.* TO 'director'@'localhost' IDENTIFIED BY 'director';
Query OK, 0 rows affected (0.020 sec)

MariaDB [(none)]>
MariaDB [(none)]>
MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.008 sec)

MariaDB [(none)]> exit
Bye
```

```
[root@localhost ~]# vim director.sh
```

```
vim director.sh
----- CONTENT OF THE FILE STARTS HERE -----
#!/bin/bash
ICINGAWEB_MODULEPATH="/usr/share/icingaweb2/modules"
REPO_URL="https://github.com/icinga/icingaweb2-module-director"
TARGET_DIR="${ICINGAWEB_MODULEPATH}/director"
MODULE_VERSION="1.7.2"
git clone "${REPO_URL}" "${TARGET_DIR}" --branch v${MODULE_VERSION}
MODULE_NAME=incubator
MODULE_VERSION=v0.5.0
REPO="https://github.com/Icinga/icingaweb2-module-${MODULE_NAME}"
MODULES_PATH="/usr/share/icingaweb2/modules"
git clone ${REPO} "${MODULES_PATH}/${MODULE_NAME}" --branch "${MODULE_VERSION}"
icingacli module enable "${MODULE_NAME}"
MODULE_NAME=ipl
MODULE_VERSION=v0.5.0
REPO="https://github.com/Icinga/icingaweb2-module-${MODULE_NAME}"
MODULES_PATH="/usr/share/icingaweb2/modules"
git clone ${REPO} "${MODULES_PATH}/${MODULE_NAME}" --branch "${MODULE_VERSION}"
icingacli module enable "${MODULE_NAME}"
MODULE_NAME=reactbundle
```

```
MODULE_VERSION=v0.7.0
REPO="https://github.com/Icinga/icingaweb2-module-${MODULE_NAME}"
MODULES_PATH="/usr/share/icingaweb2/modules"
git clone ${REPO} "${MODULES_PATH}/${MODULE_NAME}" --branch "${MODULE_VERSION}"
icingacli module enable "${MODULE_NAME}"
----- CONTENT OF THE FILE ENDS HERE -----
```

```
vim director.sh
----- CONTENT OF THE FILE STARTS HERE -----
#!/bin/bash
ICINGAWEB_MODULEPATH="/usr/share/icingaweb2/modules"
REPO_URL="https://github.com/icinga/icingaweb2-module-director"
TARGET_DIR="${ICINGAWEB_MODULEPATH}/director"
MODULE_VERSION="1.7.2"
git clone "${REPO_URL}" "${TARGET_DIR}" --branch v${MODULE_VERSION}
MODULE_NAME=incubator
MODULE_VERSION=v0.5.0
REPO="https://github.com/Icinga/icingaweb2-module-${MODULE_NAME}"
MODULES_PATH="/usr/share/icingaweb2/modules"
git clone ${REPO} "${MODULES_PATH}/${MODULE_NAME}" --branch "${MODULE_VERSION}"
icingacli module enable "${MODULE_NAME}"
MODULE_NAME=ipl
MODULE_VERSION=v0.5.0
REPO="https://github.com/Icinga/icingaweb2-module-${MODULE_NAME}"
MODULES_PATH="/usr/share/icingaweb2/modules"
git clone ${REPO} "${MODULES_PATH}/${MODULE_NAME}" --branch "${MODULE_VERSION}"
icingacli module enable "${MODULE_NAME}"
MODULE_NAME=reactbundle
MODULE_VERSION=v0.7.0
REPO="https://github.com/Icinga/icingaweb2-module-${MODULE_NAME}"
MODULES_PATH="/usr/share/icingaweb2/modules"
git clone ${REPO} "${MODULES_PATH}/${MODULE_NAME}" --branch "${MODULE_VERSION}"
icingacli module enable "${MODULE_NAME}"
----- CONTENT OF THE FILE ENDS HERE -----
~
```

[root@localhost ~]# chmod +x director.sh

[root@localhost ~]# ./director.sh

```
[root@localhost ~]#
[redacted] ./director.sh
./director.sh: line 2: -----: command not found
Cloning into '/usr/share/icingaweb2/modules/director'...

remote: Enumerating objects: 39667, done.
remote: Counting objects: 100% (60/60), done.
remote: Compressing objects: 100% (42/42), done.
remote: Total 39667 (delta 18), reused 32 (delta 13), pack-reused 39607
Receiving objects: 100% (39667/39667), 11.24 MiB | 4.04 MiB/s, done.
Resolving deltas: 100% (25529/25529), done.
Note: checking out '9c95fb8680f6f389ca24ff6e41d9002185596cb3'.
```

[root@localhost ~]# mysql -u root -p director < /usr/share/icingaweb2/modules/director/schema/mysql.sql

Enter password:

[root@localhost ~]#

[root@localhost ~]#

[root@localhost ~]# icingacli module enable director

