Secure Web

Authentication and Access Control

Table of Contents

List of Figures

1. Introduction

1.1. About Secure Web

CYAN Secure Web is Web filtering and security solution for safe and responsible browsing. Our high-performance, easy-to-use proxy server is flexible, scalable and affordable. It monitors and controls the web activities of the Internet users in a company.

1.2. About this Manual

This manual provides insight into the concepts and practical configuration of the authentication and the control of access rights of users going to the Internet. The reader is expected to have basic knowledge of the Secure Web platform, is able to access and work with the Web Admin Interface and has successful performed the initial setup steps of Secure Web, as outlined in the Getting Started Guides.

This manual is to be used with CYAN Secure Web version 2.1 and above.

For additional documentation, please see our document repository on http://www.cyan-networks.com/documentation

1.2.1. Document Conventions

⚠X	Indicates a potentially risky situation, leaving the appliance in an unusable state.
⚠	Indicates a potentially risky situation, causing misfunction of the solutions.
✎	Indicates information that is substantial for successfully configuring and using the product.
📝	Provides helpful information for the process of configuring and using the product.
💡	Provides additional information about typical scenarios and best practices.

2. Introduction

In order to manage individual access restrictions, the identity of the client is an imminent requirement for Secure Web. The rights that are available in Secure Web are managed in so-called *Profiles*, that are sets of allow and deny permissions for particular filters. Secure Web holds a list of *Profile Assignments* that define which profile should be used in the context of specific authentication information.

In short, following steps are applied to any connection in any circumstance by Secure Web:

- identify the identity of the requesting source
- find the profile that shall be applied to the connection

The identity of a requesting source may be derived from one of the following information pieces available in the request:

- The IP address of the client host
- The name of a user

Secure Web will search its Authentication Instances with this information pieces and find the IP address as a member of an IP List instance or the user as a member of an Authentication Instance. In case that none of the above pieces can be identified, Secure Web will respond with "Authentication Required" prompt to the requesting source.

A request is never allowed to pass the Secure Web engine without authentication in any form taking place. It is also not possible for a request to pass without a profile being assigned.

The following chapters will discuss the various Authentication Sources that are supported by Secure Web as well as going into detail of the various filters available in Profiles. The last chapter will connect Authentication Information to specific Profiles with help of Profile Assignments.

3. Authentication Primer

Secure Web provides numerous variants of how to find out the identity of a request in order to determine the access rules (Profile) that should be applied to this request.

Each single request will consequently follow the flow of control as shown in the following diagram:

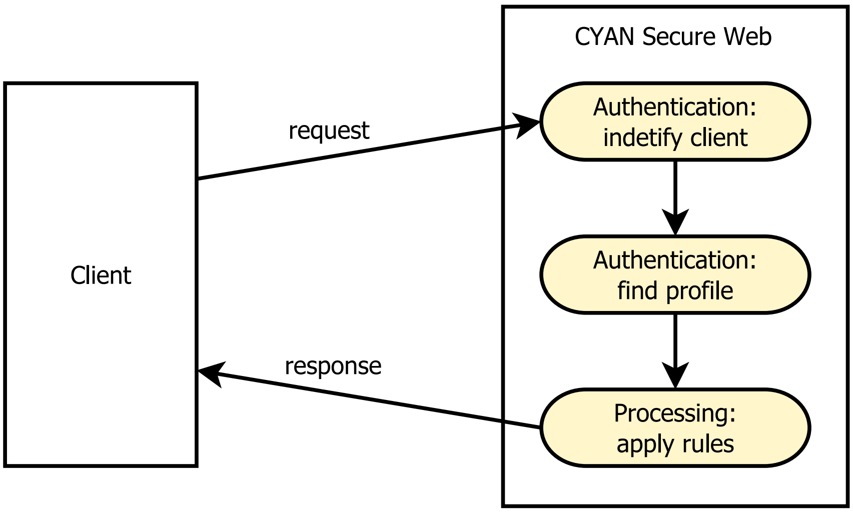


Figure 3.1. General flow of control

3.1. Terms

The following terms are the most important for understanding the concept of Authentication and its possibilities. These terms are used throughout the Secure Web interface and all corresponding documentation:

- **Authentication:** Authentication is the process of identifying a client, either by its IP address or by a user name or a token. To authenticate by IP address an IP Instance must be configured. To authenticate by user name a regular Authentication Instance must be configured.

- **Authentication Instance:** An authentication instance holds all configuration parameters to connect to a user directory. The instance must be assigned a symbolic name by which it will be referred to in subsequent actions. Authentication instances are required if the client shall be identified by a user name or a token.

- **IP Instance:** An IP Instance is a special Authentication Instance consisting of a collection of one or more IP addresses. Each IP entry must be assigned a unique symbolic name. The symbolic name will be then used as the identity of the client. IP Instances are needed when clients should be identified by their IP addresses.

- **Authentication Source:** An Authentication Source is an external service that provides all necessary data that Secure Web needs to authenticate a client (for example, user name, password, group membership). This can be a database, LDAP directory or Microsoft Active Directory.

- **Authentication Method:** An Authentication Method determines the type of authentication scheme that is provided by an Authentication Source. There are two Authentication Methods supported by Secure Web:

  - Basic Authentication

  - NTLM Authentication (also referred as Transparent Authentication or Challenge Response)

- **Basic Authentication:** Basic Authentication is an Authentication Method where a client must provide a user name and a password to be identified. Therefore it is a type of non-transparent Authentication Method. The technical details of Basic Authentication are defined within the HTTP protocol. Basic Authentication is provided by the following Authentication Sources:
  - LDAP server
  - SQL database
  - Microsoft Active Directory
- **NTLM Authentication:** NTLM Authentication is an Authentication Method that has been developed by Microsoft. The user does not have to provide any user name or password, but is automatically authenticated by a client program (for example, the browser) by going through a so called challenge/response mechanism. NTLM Authentication is a type of transparent Authentication Method. Only Windows clients are able to use NTLM Authentication. NTLM Authentication is provided by the following Authentication Sources:
  - Microsoft Active Directory
- **Identification:** As a result from Authentication, Identification uniquely identifies the client. Depending on the Authentication Source, this can be a username or an IP address of the client. The Identification is then used as the primary source for Profile Assignment.
- **Secure Authentication Service:** It performs external authentication, i.e. it functions as an intermediate service between Secure Web and an Authentication Source. This includes Basic Authentication against an LDAP server, SQL database, or Microsoft Active Directory and NTLM Authentication against a Microsoft Active Directory. Secure Authentication Service may be referred to in short as "sauth".
- **Secure Web:** It performs internal authentication, i.e. it contacts Authentication Sources by itself, if necessary. This includes identification by IP address and Basic Authentication against an LDAP server or SQL database. sweb is the short name of the Secure Web proxy service.

3.2. Authentication Methods

An authentication method refers to a kind of method used to determine the requesting source identity:

- **IP**: the IP address of the requesting source is used as the identity. Secure Web searches for the first match in the list of IP instances. The symbolic name of the instance found is used as the identification.
- **Basic**: the proxy requests a username and a password from the web client. The web client will as a result of this request popup a window asking for the authentication credentials, and will afterwards provide these credentials to the proxy, which verifies its validity.
- **NTLM (Challenge/Response)**: the web client will be queried using challenge information which is provided by the Active Directory domain controller. The response provided from the web client to the proxy server will in turn be verified with AD. No passwords, only tokens are sent back and forth. If this exchange processes successfully the proxy will learn about the user that is logged into the client system. The whole process is transparent (not recognized by the end user).

3.3. Authentication Instances

An authentication instance holds all configuration parameters to connect to a user directory. The instance will get assigned a symbolic name by which it will be referred to in subsequent actions.

Authentication Instances can be connected to various Authentication Sources:

3.3.1. IP List

IP addresses are available in all requests from the clients. This Authentication Instance provides a possibility to identify the client based on the IP address of the host. The IP instance maps single IP addresses or pools of IP addresses to symbolic names and does not connect to any external Authentication Source.

A typical use case for the IP address authentication is in the context of servers available in the network. Servers usually have a statically assigned IP address and run automatic processes which never interact with a user, for which reason nobody can fill out a login form.

3.3.2. Microsoft Active Directory

Secure Web can be integrated with your Domain / Active Directory server for the purpose of identifying and authenticating the users that want to gain access to the Internet. Having Secure Web connected to the Active Directory also opens the option to transparently authenticate Windows users using the NTLM challenge/response method.

Special software called "Secure Authentication Service" needs to be installed on a domain controller or on any Windows machine, which is already a member of the domain. The Secure Web server will communicate with the Authentication Service in order to query user and group information and to verify authentication information.

Using Windows Authentication Service provides full ability to retrieve user and group information. However, the Authentication Service requires to be installed on a separate machine. On the machine that runs the Authentication Service the NTLM Authentication will not work, i.e. it is not advisable to use this machine as a user workstation too.

3.3.3. Novell eDirectory

Authentication against Novell eDirectory is done via LDAP and supports both IP authentication and Basic authentication.

3.3.4. LDAP

The Lightweight Directory Access Protocol (defined in RFC 4511), or in short LDAP provides access to distributed directory services that act in accordance with X.500 data and service models. These protocol elements are base on those described in the X.500 Directory Access Protocol (DAP). There are examples of X.500 based directory services like Microsoft's Active Directory included in Windows Server operating system and Novell's eDirectory implementation.

In our case Secure Web uses the LDAP protocol to query user and group information as well as to verify username and password information against user directory.

This Authentication Instance type supports Basic authentication method. Web browser will present a pop-up window to the user in reaction to the proxy authentication request, asking for the name and the password of the user.

Using LDAP Authentication Instance is useful in case there is an existing LDAP server in the network already, and the users and groups which are managed in this server shall be reused in Secure Web to authenticate themselves. The user and groups can be then assigned individual profiles to control the access permissions to the Internet.

3.3.5. SQL Database

Secure Web can be connected to an SQL database interface, which provides an easy way to perform Basic Authentication against an existing or newly deployed user database. Following database engines are supported by Secure Web:

- MySQL
- PostgreSQL
- SQLite3

Please note that group membership is not supported in SQL Database Authentication Instances. Authentication against SQL databases is only available on CYAN Appliances, CYAN Virtual Appliances and all Linux based products. The Microsoft Windows platform does not support authentication against database systems.

3.4. Authentication Infrastructure

Secure Web supports two approaches to establish the communication with an Authentication Source:

- Internal methods built into the Secure Web (identified in the following table as "sweb")
- Utilizing the external Secure Authentication Service (identified in the following table as "sauth")

The following table outlines the correlation between Authentication Instance, Authentication Source and Authentication Method:

Authentication Source	Authentication Instance type	internal (sweb)	external (sauth)	Authentication Method
IP Address	IP List	X		IP Address
LDAP Server	LDAP	X	X	Basic
SQL Database	Database	X	X (Linux only)	Basic
Active Directory	Active Directory		X	Basic,NTLM

3.4.1. Built-in Authentication Methods

The following diagram illustrates the setup where the built-in Authentication Instances are used for authentication (LDAP and SQL Database):

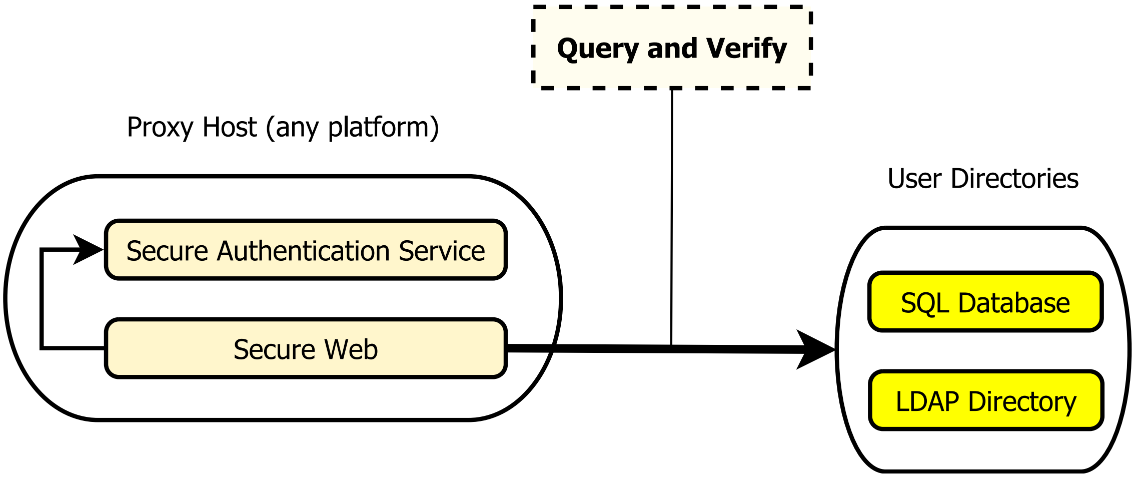


Figure 3.2. Communication channels using built-in Authentication Instances

In order to communicate with the SQL Database or the LDAP directory, Secure Web established TCP connection to the server hosting the User Directory. In case a firewall separates the Secure

Web from the User Directory, TCP communication must be allowed on the firewall on necessary ports.

The typical TCP ports in use are the following ones:

- LDAP: 389
- MySQL Database: 3306
- PostgreSQL Database: 5432

Following list summarizes support for various authentication features:

- Authentication variants: Basic
- Group membership resolution: supported
- Nested group membership resolution: not supported

Note that Group membership resolution support shows if an Authentication Source is capable of having user groups and return the group membership information. Nested group membership resolution shows support for having nested groups inside another group.

3.4.2. Proxy on Linux connected to Active Directory

The following diagram illustrates the setup where the Microsoft Active Directory Authentication Instances is used for authentication on a Linux host, using the Secure Authentication Service running on a Windows host:

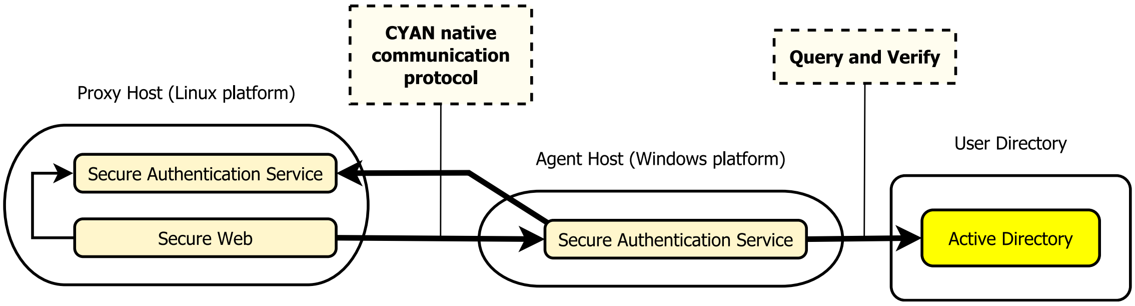

CYAN native communication protocol
Query and Verify
Proxy Host (Linux platform)
Secure Authentication Service
Secure Web
Agent Host (Windows platform)
Secure Authentication Service
User Directory
Active Directory

Figure 3.3. Communication channels utilizing the Authentication Service from Linux

The Secure Authentication Service is installed on a Windows platform. That provides an access to the Microsoft Active Directory using the native Windows API, enabling the full functional set made available by Microsoft.

The communication between Secure Web and Secure Authentication Service is done via a TCP connection. The Secure Authentication Service must be also able to connect to the Secure Configuration Service in order to retrieve the centrally stored configuration parameters. In case a firewall is involved in the communication, a TCP traffic must be allowed on the firewall on necessary ports.

The TCP ports in use are the following ones:

- Secure Configuration Service: 9991
- Secure Authentication Service: 9995

Following list summarizes support for various authentication features:

- Authentication variants: Basic, NTLM
- Group membership resolution: supported
- Nested group membership resolution: supported

3.4.3. Proxy on Windows connected to Active Directory

The following diagram shows the setup of the Secure Web together with Secure Authentication Service, both on a Windows host:

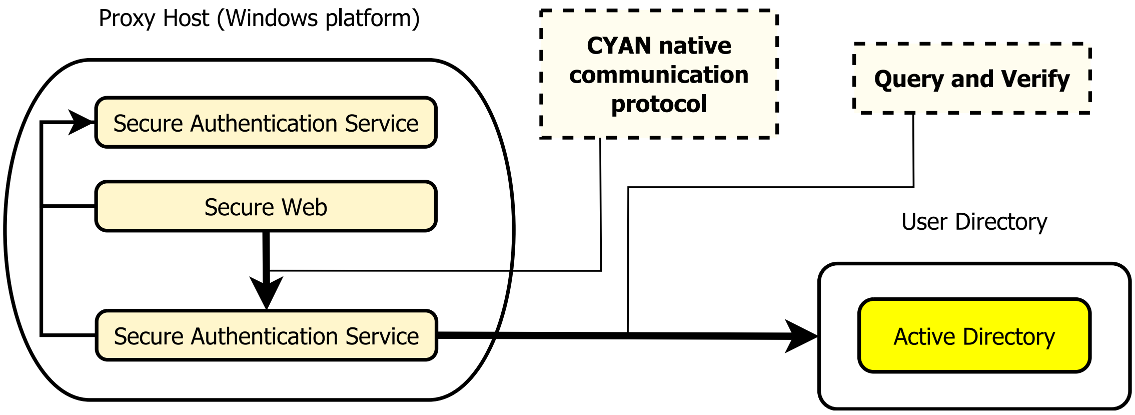

Proxy Host (Windows platform)
CYAN native communication protocol
Query and Verify
Secure Authentication Service
Secure Web
User Directory
Secure Authentication Service
Active Directory

Figure 3.4. Communication channels utilizing the Authentication Service from Windows

In this setup all the communication of the CYAN services is carried out on the localhost, so no additional firewall rules are necessary. The communication is based on the native Windows API. The host machine must be a member of a Windows domain a must be assigned sufficient access permission to carry out the query and verification tasks.

Following list summarizes support for various authentication features:

- Authentication variants: Basic, NTLM
- Group membership resolution: supported
- Nested group membership resolution: supported

3.5. Authentication and Proxy Modes

The Secure Web can be used in two different modes:

- Dedicated proxy server - In this mode a client application has to be explicitly configured to use the Secure Web, i.e. the application is aware that it needs to interact with the proxy server in order to connect to targeted servers.
- Transparent proxy server - The Secure Web processes all the necessary network traffic, which is redirected to its ports using firewalls or routing techniques. The client application is not aware of the existence of any proxy server.

Web browsers support web proxies in configuration of the network settings. As soon as a browser is configured to use the Secure Web as a proxy server, the browser is aware of the existence of a proxy on the network and will react to special HTTP status codes (for example 407 Proxy Authentication Required). If there was no proxy server configured, the browser would not know about a transparent proxy and would not react to these status codes as a security measure. This behavior prevents a situation, where a malicious web server would ask for a "407 Proxy Authentication" and would try to trick the user into thinking, that he/she is authenticating against a company proxy (but in fact send the credentials in clear text across the Internet to a foreign server).

For reasons state above, a web browser will not do proxy authentication if there is no proxy configured.

Other protocols supported by the Secure Web (FTP, POP3, IMAP) have by design no support for proxy servers. The use of such protocols with a proxy server requires extending the login information of these protocols. This way a user can authenticate against both the Secure Web and the targeted protocol destination.

The following table provides an overview of the authentication abilities depending on the proxy mode being used:

Protocol	Authentication in Dedicated Mode	Authentication in Transparent Mode
HTTP, HTTPS	Yes	No
FTP	Yes*	Yes*
POP3, POP3S	Yes*	Yes*
IMAP, IMAPS	Yes*	Yes*

* Extended authentication information is required.

3.5.1. Extended authentication for FTP

For FTP is not necessary to actually add anything to the login information. Most of the FTP clients have firewall or proxy settings and provide a set of schemes implemented to tell the firewall or proxy all the required information. It is necessary just to select one already implemented in Secure Mail Proxy from the following list:

```
USER Remote_User@Remote_Host Proxy_User
PASS Remote_Password
ACCT Proxy_Password

USER Remote_User@Proxy_User@Remote_Host
PASS Remote_Password@Proxy_Password

USER Proxy_User@Remote_Host
PASS Proxy_Password

OPEN Remote_Host
USER Remote_User
PASS Remote_Pass
```

3.5.2. Extended authentication for mailing protocols

Secure Mail Proxy is listening on the following ports:

- POP3: 8110

- POP3S: 8910
- IMAP4: 8143
- IMAP4S: 8943

You can choose out of three different authentication schemes. The Secure Mail Proxy will automatically detect the following ones:

1. Proxy authentication, server authentication and server name (full proxy with proxy authentication):

   ```
   User: #<proxy-user>#<remote-user>#<remote-server>
   Pass: #<proxy-password>#<remote-password>
   ```

2. Server authentication and server name (full proxy without proxy authentication, for example when IP List Authentication Instance is used for authentication):

   ```
   User: #<remote-user>#<remote-server>
   Pass: #<remote-password>
   ```

3. Server authentication only (transparent proxy):

   ```
   User: <remote-user>
   Pass: <remote-pass>
   ```

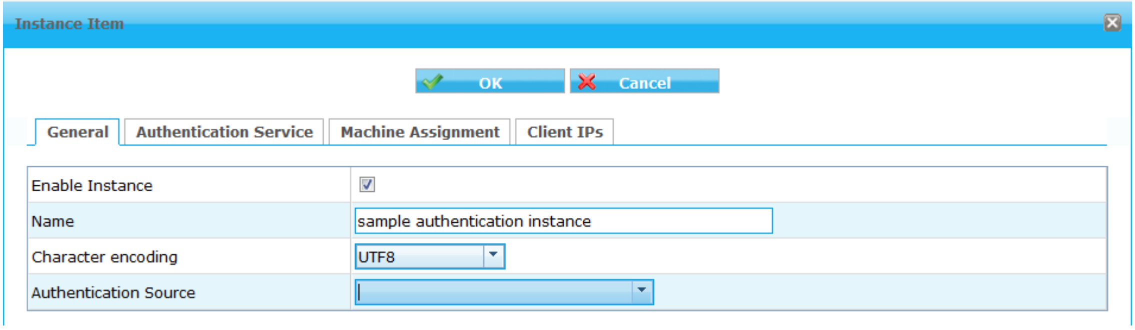
In the previous examples substitute *<proxy-user>* and *<proxy-password>* with the proxy Username and Password, *<remote-server>*, *<remote-user>* and *<remote-password>* with the IP or URL address of the remote server and the login credentials.

The delimiter ('#' in the previous examples) is determined from the first character used. It can be chosen freely and will be automatically detected, if the first character is none of the characters `A-Z`, `a-z` and `0-9`.

3.6. Setup of Authentication Instances

An Authentication Instances can be configured through the Web Admin Interface via *Services/Authentication/Instances* menu entry.

In the list of instances, right-click an empty spot and choose *Add item…* to create a new instance. A window opens with the configuration options for the new Authentication Instance, as you can see in the following figure:



Enable Instance	☑
Name	sample authentication instance
Character encoding	UTF8
Authentication Source	

Figure 3.5. Adding a new Authentication Instance

Visibility of certain tabs depends on the Authentication Source selected.

CYAN Secure Web also supports the use of multiple Authentication Instances in parallel. Close care needs to be taken about the interaction of multiple Authentication Instances, in order to avoid mutual exclusions of functions or unpredictable behavior of the overall system.

Authentication Instances of type IP List get always processed first, then all the other Authentication Instances in order as they appear in the list. A user gets always validated against the first Authentication Instance where is he found. For that reason we strongly recommend to keep usernames unique across all the Authentication Instances to prevent unexpected login failures.

When searching for matching profile, the Secure Web follows the following flowchart diagram:

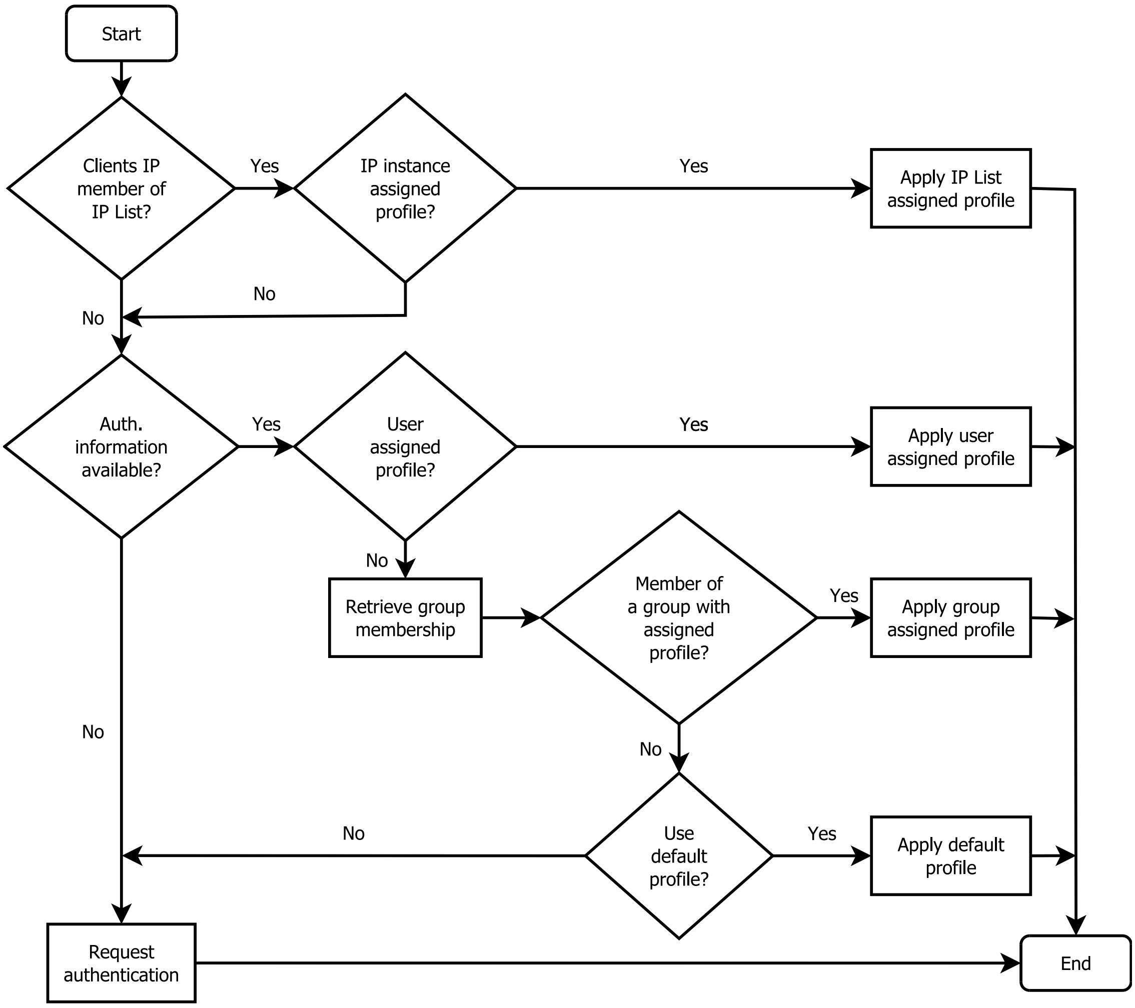

Start
Clients IP member of IP List?
Yes
IP instance assigned profile?
Yes
Apply IP List assigned profile
No
No
Auth. information available?
Yes
User assigned profile?
Yes
Apply user assigned profile
No
Retrieve group membership
Member of a group with assigned profile?
Yes
Apply group assigned profile
No
No
No
Use default profile?
Yes
Apply default profile
Request authentication
End

Figure 3.6. Diagram of searching for a matching profile

3.6.1. General configuration

General

- **Enable Instance** - Should be *checked* to enable the instance.
- **Name** - Freely chosen name of the Authentication Instance which will be referred to in subsequent actions.
- **Character encoding** - Needs to be set to the character encoding used for encoding special characters (umlauts, sharp-s) on clients. For most environments, UTF8 will be the proper

choice. Russian environments may set this to one of the Russian encodings used on clients, like KOI8R or CP1251.

- **Authentication Source** - Defines the Authentication Source to be used in this Authentication Instance. The specific settings that open up if you choose any of the available Sources here will be covered in the following chapters.

Authentication Service

- **Use Authentication Service** - Needs to be enabled if an external CYAN Secure Authentication Service should be used for processing authentication. This is needed for authentication against Microsoft Active Directory from a Secure Web Appliance or Linux installation. The Authentication Service must be installed on a Windows member server.
- **Primary host** - The IP address of the external Secure Authentication Server
- **Primary port** - The port of the external Secure Authentication Server, usually 9995.
- **Backup host** - The IP address of another Secure Authentication Server acting as a backup server.
- **Backup port** - The port of the backup Secure Authentication Server, usually 9995.
- **Alternate hosts** - If enabled, sends authentication requests to both Secure Authentication Server and balances the load across both of the machines.

> Authentication Service configuration is not available if *"IP List"* Authentication Instance is selected. Authentication against an IP list is always performed internally in Secure Web and is never send to an external Secure Authentication Service.

Machine assignments

- **Assign Instance to specific machines** - If checked, restricts the initialization of this Authentication Instance to the selected Secure Web Appliances. This is mainly used in distributed environments where multiple Secure Web Appliances are managed centrally with different Authentication Sources on each location.

Client IPs

- **Use Instance for specific Client IPs** - If unchecked, applies this Authentication Instance to request from all clients. If checked, a list of clients for which this Instance should be used for authentication, can be configured. This is needed for environments where network segments, distinguishable through IP network addresses, need to be authenticated against different Authentication Sources (e.g. Network 10.1.0.0/16 # LDAP, Network 172.16.1.0/24 # Database).

These settings are common across all Authentication Sources and work the same, regardless of what actual Authentication Source will be configured later. The specific settings will be discussed in the following chapters.

> ⚠ There are few important aspects in the Authentication Instance configuration which knowledge may be crucial. Please pay good attention to the list below this box.

- Only activated Authentication Methods can be used by clients.
- IP List Authentication Instances are always checked first. If a client's IP address has been matched by an IP List, no other Authentication Instances will be checked.

- The order of Authentication Instances as they appear in the Web Admin Interface is important. They are checked from the top to the bottom. If a client has been authenticated by a certain Authentication Instance, no other Authentication Instances are checked. The order of an Authentication Instance can be changed in a drag-and-drop manner in the list.

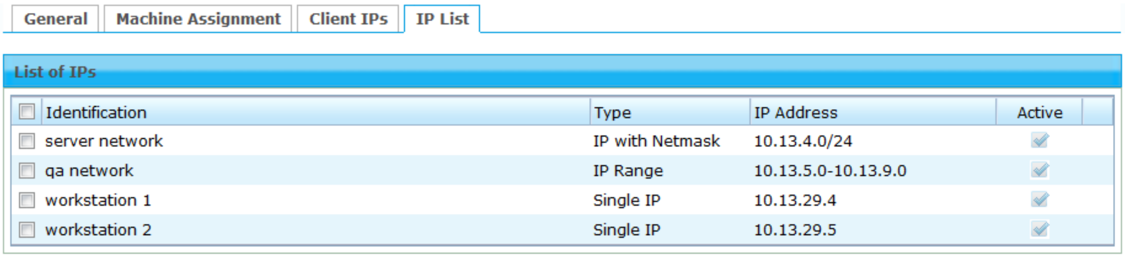
- Web browser does not fall back from NTLM Authentication to Basic Authentication. If both methods are activated, the Secure Web will provide both methods to the client. A Windows client will choose the stronger NTLM Authentication method. If this authentication fails, the web browser will not try it authenticate using the Basic authentication afterwards.

Any Authentication Instance can be tested from the list of Authentication Instances by selecting the Authentication Instance (via the check box), right clicking it and selecting *"Test Instance"*.

3.6.2. IP List

This section assumes you have already read Section 3.6, “Setup of Authentication Instances”. Configuration of an IP List Authentication Instance type can be done in the *"IP List"* tab. In a default configuration there is one IP List instance named “IP List” already present. This Authentication Instance comes with one IP list named “global” representing “the world” (0.0.0.0/0) and matches all IP addresses.

A new list item can be added by right clicking the list and selecting *"Add item"*. Example of such a list follows:



☐ Identification	Type	IP Address	Active
☐ server network	IP with Netmask	10.13.4.0/24	✓
☐ qa network	IP Range	10.13.5.0-10.13.9.0	✓
☐ workstation 1	Single IP	10.13.29.4	✓
☐ workstation 2	Single IP	10.13.29.5	✓

Figure 3.7. IP List Authentication Instance example

Following list of information has to be supplied in order to add a new IP list entry:

- **Identification** - A name that should be applied to the entry.

- **Type** - Specifies the type of the following entry. Possible options are:

  - **Single IP** - This entry will match only one specified IP address.

  - **IP with Netmask** - This entry will match a set of IP addresses with a common prefix. The network identification has to be supplied either as a network prefix (e.g. 192.168.1.0/24 for a network 192.168.1.0 and network mask 255.255.255.0), or as a network mask (192.168.1.0/255.255.255.0).

  - **IP Range** - This entry will match a range of IP addresses specified by two IP addresses separated by a dash sign (e.g. 192.168.1.1-192.168.1.254).

- **IP Address** - Actual IP address to be matched (or a range of IP addresses). Depends on the *Type* field.

The IP List Authentication instance matches first, prior to all other instances. Whenever the IP address of the client host is matched against an entry in the IP List Authentication Instance, the request is processed and no additional authentication will be requested.

3.6.3. Microsoft Active Directory

This section assumes you have already read Section 3.6, “Setup of Authentication Instances”. Configuration of a Microsoft Active Directory Authentication Instance type can be done in the *"Microsoft Active Directory"* tab. Authentication against Microsoft Active Directory on Secure Web Appliances must be done with help of an external Secure Authentication Service.

The Authentication Source supports both Basic Authentication as well as NTLM.

- **NetBIOS Domain** - Contains a comma separated list of NetBIOS names of your domain (eg CYAN, FOOBAR).

- **Use prefix for multiple Domains** - Needed in multi domain environments if Basic Authentication should be performed and usernames are prefixed by their domain name (e.g. CYAN\user).

> Please make sure that the NetBIOS domain of your domain is configured. The NetBIOS name is at maximum 16 characters long and is typically written in uppercase only letters (e.g. CYAN). The DNS name of the domain looks like an Internet domain (e.g. cyan.local) and must not be used here.

The Secure Authentication Service for Windows is used as an agent application, which is installed on a Windows system and communicates with the Active Directory server for the purpose of querying user and group information and to verify user authentication. The Secure Web utilizes the Authentication Service as an agent for its communication with Active Directory. The communication between the Secure Web and the Authentication Service is established via TCP connections. All the interactions can be seen in Figure 3.3, “Communication channels utilizing the Authentication Service from Linux”.

In order to setup Secure Web authentication utilizing the Authentication Service for Windows, following steps need to be completed:

1. Install the Secure Authentication Service on a Windows host

2. Configure the Authentication Service

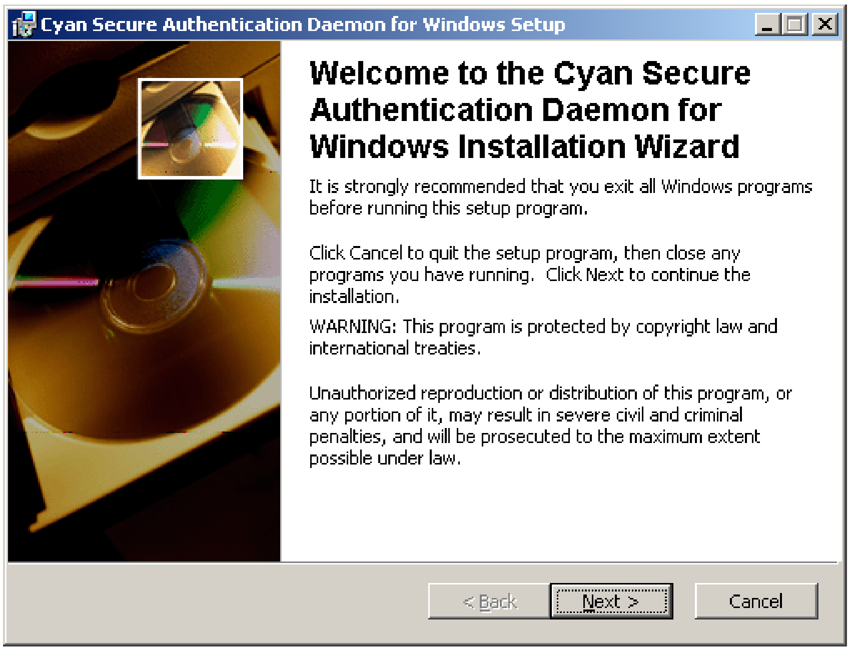
3. Create an Authentication Instance and connect to the Authentication Service

3.6.3.1. Installing the Secure Authentication Service on a Windows host

The Secure Authentication Service for Windows can be installed either on one of your domain controllers (primary, backup) or on any Windows machine that is a member of the domain. The software will be installed and run as a system service.

> The latest copy of the Secure Authentication Service for Windows can be obtained from our homepage:
>
> http://www.cyan-networks.com/index.php/en/downloads/secure-authentication-daemon

In order to install the Authentication Service start the EXE installer you have downloaded and follow the instructions on the screen:



Welcome to the Cyan Secure Authentication Daemon for Windows Installation Wizard

It is strongly recommended that you exit all Windows programs before running this setup program.

Click Cancel to quit the setup program, then close any programs you have running. Click Next to continue the installation.

WARNING: This program is protected by copyright law and international treaties.

Unauthorized reproduction or distribution of this program, or any portion of it, may result in severe civil and criminal penalties, and will be prosecuted to the maximum extent possible under law.

Figure 3.8. Adding a new Authentication Instance

After completion of the installation process, the CYAN Secure Authentication Manager entry can be found in the start menu of your system under *"Cyan Networks/sauth_mg.exe"*.

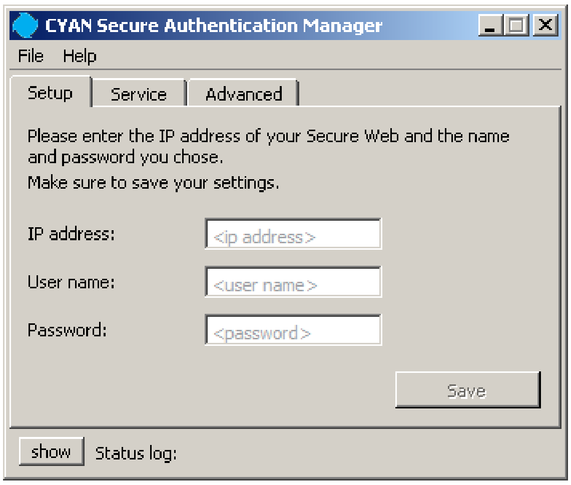
CYAN Secure Authentication Service and Secure Authentication Manager for Windows depend on Microsoft Visual C++ Runtime libraries. Usually the required libraries are installed on a Windows Server already. If not, please download and install the installation package from Microsoft via the following link:

http://www.microsoft.com/en-us/download/details.aspx?id=29

3.6.3.2. Configuration of the Secure Authentication Service

Start the CYAN Secure Authentication Manager in order to setup the connection parameters, i.e. the IP address, login information and port number of the Secure Configuration Service (typically the same IP address as the one of you Appliance and port number 9992).

The IP address and login information can be set in the *Setup* tab as in the following figure:



Please enter the IP address of your Secure Web and the name and password you chose.
Make sure to save your settings.

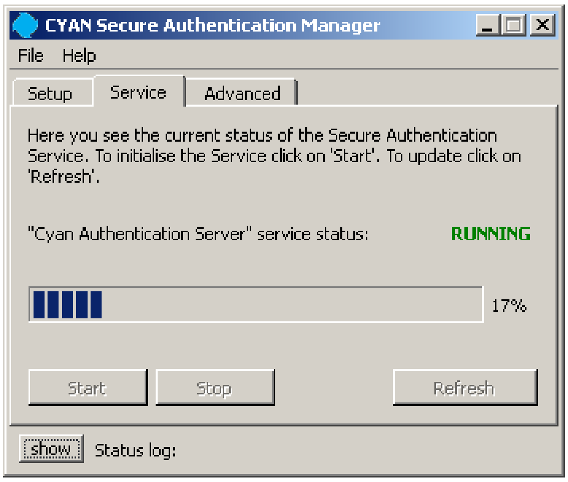
IP address: <ip address>

User name: <user name>

Password: <password>

Figure 3.9. Setting up IP address and login information

If you filled in all the information correctly and pressed the *Save* button, you can switch to the *Service* tab (see Figure 3.10, “Service status”) and verify the status is *RUNNING*.



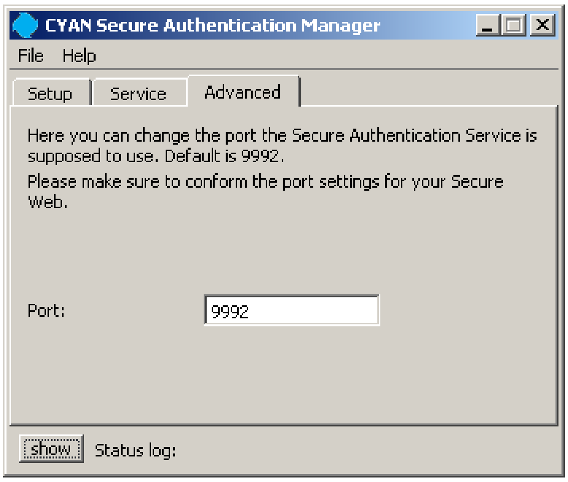
Here you see the current status of the Secure Authentication Service. To initialise the Service click on 'Start'. To update click on 'Refresh'.

"Cyan Authentication Server" service status: RUNNING

17%

Figure 3.10. Service status

It probably will not be necessary to change the default port number. The change can be done in the tab *Advanced*, as in the following figure:



Here you can change the port the Secure Authentication Service is supposed to use. Default is 9992.

Please make sure to conform the port settings for your Secure Web.

Port: 9992

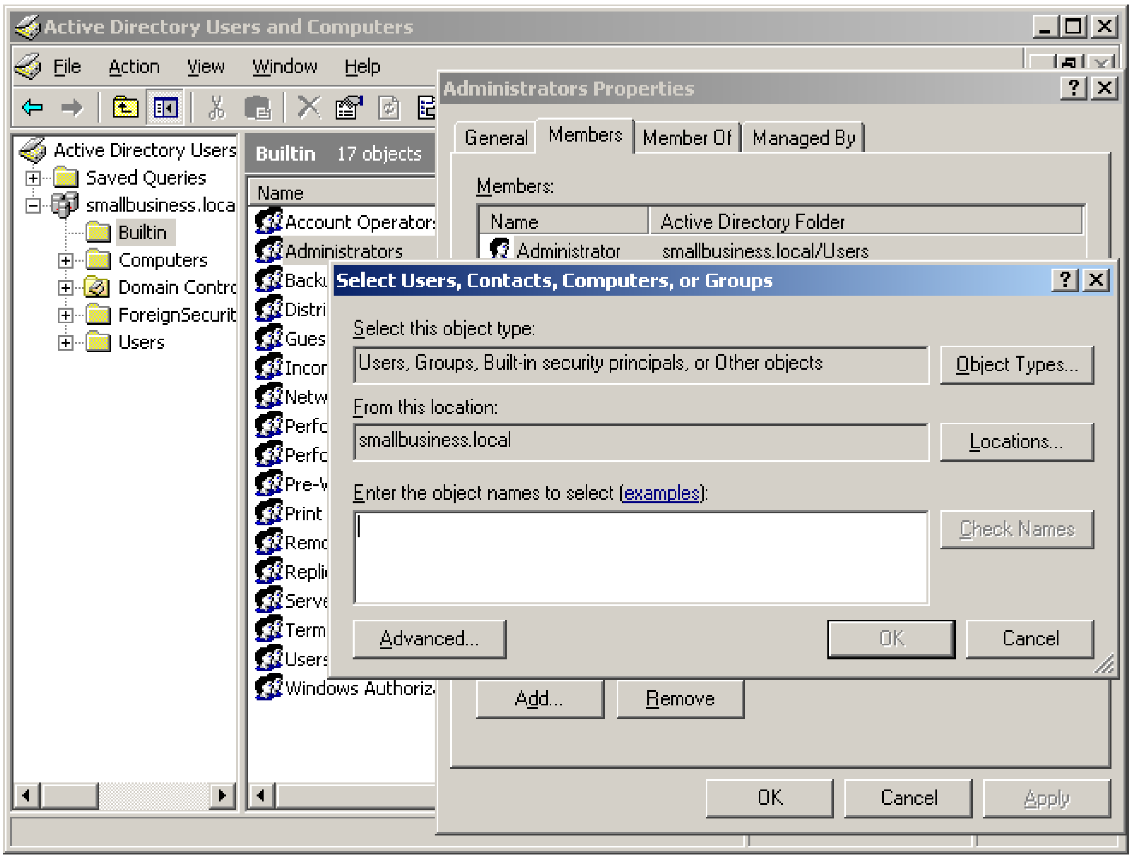
Figure 3.11. Setting up port number

3.6.3.3. Setting Active Directory permissions for the Secure Authentication Service

Provided the Secure Authentication Service is installed on a domain controller, the permissions fort the system service will be sufficient for all required operations.

In case that Secure Authentication Service is installed on any Windows host that is a member of the domain, the system service will most probably not have sufficient permissions. In order to provide the proper level of access rights, various approaches can be chosen. One of them is adding the computer to the administrators group of the domain.

The following figure shows the *"Active Directory Users and Computers"* configuration dialog. In here you can add the computer to the group *Administrator* in the *Members* tab by pressing the *Add* button. Make sure in *Object Types…* is also selected value *Computers*.



Members:

Name	Active Directory Folder
Administrator	smallbusiness.local/Users

Select this object type: Users, Groups, Built-in security principals, or Other objects

From this location: smallbusiness.local

Figure 3.12. Setting up Active Directory settings

At this point it should be possible to completely configure the Authentication Instance (please review Section 3.6.3, “Microsoft Active Directory”) and connect to the Authentication Service (please review Section 3.6.1, “General configuration”).

3.6.4. Novell eDirectory

This section assumes you have already read Section 3.6, “Setup of Authentication Instances”. Configuration of a Novell eDirectory Authentication Instance type can be done in the *"Novell eDirectory"* tab. Authentication against Novell eDirectory is done via LDAP and supports both authentication based on the client IP as well as the Basic Authentication. In order to successfully setup the Authentication Instance all of the following information has to be supplied:

- **Host** - A hostname or an IP address of an eDirectory LDAP server.
- **LDAP Bind DN** - The Bind DN used to retrieve information via LDAP.
- **LDAP Bind Password** - The Bind Password used to retrieve information via LDAP.
- **LDAP Base DN** - The Base DN for user and group search operations.

3.6.5. Standard LDAP Server

Authentication against any directory server following the standardized LDAP protocol can be used for Basic Authentication. This section assumes you have already read Section 3.6, “Setup of Authentication Instances”. Configuration of an LDAP Authentication Instance type can be done in the *"LDAP"* tab. All the following information needs to be supplied:

- **Host** - A hostname or and IP address of an LDAP server.

- **Port** - The network port used for connection to the LDAP server. The default value of "389" is the default TCP port for LDAP.

- **Backup host** - An alternative hostname or IP address of a secondary LDAP server which will be used if the primary server is unavailable.

- **Backup port** - The network port used for connection to the secondary LDAP server.

- **LDAP Bind Method** - Sets the authentication mode used for binding to the LDAP tree. If no authentication is necessary to bind to the LDAP directory, *LDAP_AUTH_NONE* should be used and *LDAP Bind DN* and *LDAP Bind Password* can be left empty.

- **LDAP Bind DN** - The distinguished name (DN) specifies the location of the (administrative) user which is used by the service to authenticate the connection. This user must have sufficient rights in order to query user and group lists as well as to perform login verification on behalf of the actual LDAP user that wants to access the Internet. A typical bind DN entry look like `cn=admin,cn=User,ou=cyan-networks`. This example specifies the path in the directory hierarchy using the domain component (DC) and a common name (CN) attributes.

- **LDAP Bind Password** - The password of the LDAP account (DN, distinguished name) to be used for retrieving information from the LDAP server.

- **LDAP Base DN** - The path in the directory hierarchy where Secure Web will start searching for users and groups. Note that the entry point for searches must provide access to both the user and the group information available in the directory.

- **LDAP User Attribute** - The name of the attribute that contains a user name.

- **LDAP User OC** - The object class of user objects in the LDAP tree.

- **Enable Groups** - Enables or disables the retrieval of groups and group membership from the LDAP directory.

- **LDAP Group Attribute** - The name of the attribute that contains a group name.

- **LDAP Group OC** - The object class of group objects in the LDAP tree.

- **LDAP Group Member** - The name of an attribute that contains a list of user names that are members of a group.

- **Use Paged Results** - If enabled, pages the result set of user and group list operations to avoid an excess number of objects to be returned from the LDAP server. Some servers limit the number of objects that may be retrieved in a single operation and thus paging needs to be performed to consequently retrieve the whole list.

- **Result Page Size** - If *Use Paged Results* is enabled, defines the size of one result set from the LDAP server. For example, Microsoft Active Directory in its default security configuration allows a maximum page size of 1000.

- **Search Filter for User/Group List** - Contains and optional LDAP filter that is applied when searching for user and group objects.

- **Pre Configuration** button - Pre-fills the LDAP instance parameters with values suitable for connectivity to Microsoft Active Directory. Placeholders for values to be filled in by the administrator will be replaced with the proper values for LDAP Bind DN and LDAP Base DN. Please note that any previously inserted data will be overwritten and lost.

Due to the large amount of different LDAP servers available and no standardized way of saving user and group information in a directory, the values for a correct configuration of the LDAP instances vary from installation to installation. Please consult your LDAP directory administrator for the correct way to access the directory.

3.6.5.1. Example - general configuration

The following examples shall explain the specification of the base DN and the consequences for finding users and groups in more detail. Let’s assume following simple hierarchy implemented in the directory:

```
o=cyan-networks
    ou=sales
    ou=support
    ou=development
```

In case the search mask for the structure above is specified like the following

```
ou=sales,o=cyan-networks
```

the query will find only users that are created in the organizational unit *sales*, users from the organizational units *support* and *development* will be invisible to the Secure Web authentication.

The search mask for the structure above which will find all users from all the organizational units therefore must be specified as

```
o=cyan-networks
```

3.6.5.2. Example - LDAP and Microsoft Active Directory

The structure of a standard installation of Microsoft’s Active Directory looks like the following:

```
dc=com
    dc=cyan-networks
        cn=users
            cn=peter
            cn=paul
            cn=mary
            cn=sales
```

The hierarchy is specified using domain components (DC). The *users* element holds all the user and group objects. A group in an LDAP directory is an object that holds a list of users, which are members of the corresponding group. A proper search mask will look like the following:

```
cn=users,dc=cyan-networks,dc=com
```

In order to identify the user and group objects and to retrieve the group memberships from the Active Directory, following attribute specifications are required:

- **LDAP User Attribute:** `sAMAccountName`
- **LDAP User OC:** `user`
- **LDAP Group Attribute:** `cn`
- **LDAP Group OC:** `group`
- **LDAP Group Member:** `member`

3.6.6. SQL Database

This section assumes you have already read Section 3.6, “Setup of Authentication Instances”. Configuration of an SQL Database Authentication Instance type can be done in the *"SQL Database"* tab. All database instances share the following settings:

- **Authentication Query** - Defines an SQL query to be used to check the username and password against values stored in the database. As soon as a non-empty result set is returned, the user is assumed to be authenticated. The query must contain the keywords %USERNAME% and %PASSWORD% which will be replaced by the respective values during authentication (for example, `SELECT name FROM users WHERE name=%%USERNAME%% and pass=encrypt(%%PASSWORD%%)`).

- **Enumeration Query** - Defines an SQL query that returns the usernames stored in the database (for example, `SELECT name FROM users`).

Database engine specific settings are covered in the chapters below.

3.6.6.1. MySQL

- **Host** - A hostname or an IP address of a MySQL server.

- **Port** - The network port to be used to connect to the MySQL server. The default value "3306" is the default TCP port for MySQL databases.

- **Database Name** - The name of the database to connect to.

- **Database User** - The user account to be used to connect to the database.

- **Database Password** - The password of the user used to connect to the database.

> Please verify that your MySQL server is configured to allowed connections via TCP and the supplied *Database User* is allowed to connect from all Secure Web IP addresses.

3.6.6.2. PostgreSQL Database

- **Host** - A hostname or an IP address of a PostgreSQL server.

- **Port** - The network port to be used to connect to the PostgreSQL server. The default value "5432" is the default TCP port for PostgreSQL databases.

- **Database Name** - The name of the database to connect to.

- **Database User** - The user account to be used to connect to the database.

- **Database Password** - The password of the user used to connect to the database.

> Please verify that your PostgreSQL server is configured to allowed connections via TCP and the supplied *Database User* is allowed to connect from all Secure Web IP addresses.

3.6.6.3. SQLite3 Database

- **Database Path** - A file path to the SQLite3 database file on the Secure Web local file system.

> We do not advise to configure authentication against SQLite3 databases in a cluster environment. In cluster environments, the database files need to be published and kept

synchronised across all cluster members, which is not provided natively by Secure Web and needs a manual configuration.

4. Profiles Primer

Secure Web features numerous filters and security modules that may be applied to specific users, groups and IP networks. This set of filters is grouped in a profile. Every request that is processed through Secure Web must have a profile applied.

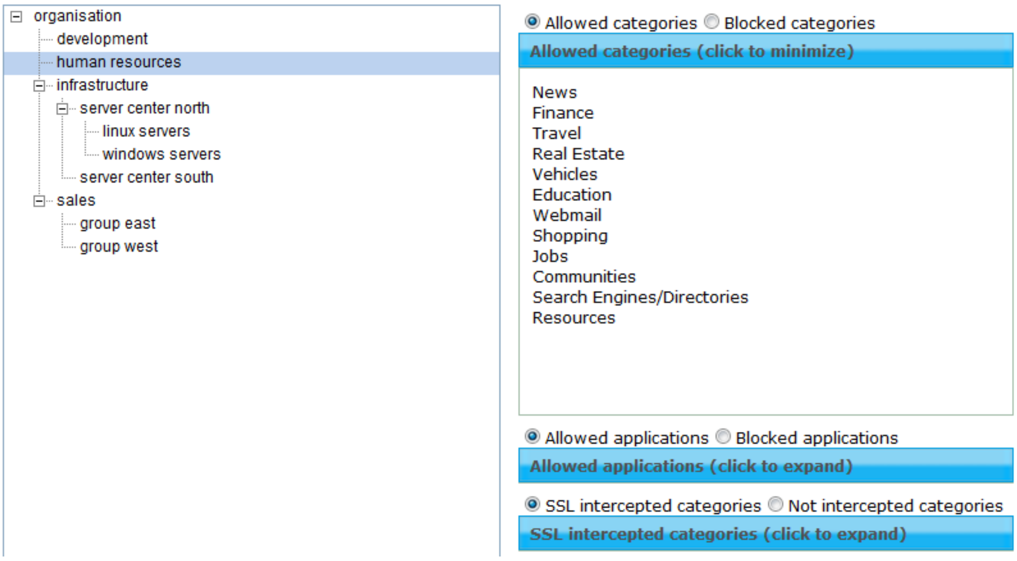
> All requests passing the Secure Web proxy engine must map to a specific profile. There are no requests allowed to pass the engine without a profile being assigned.

4.1. Profile Tree

Profiles are stored in a tree to minimize the administrative efforts. The tree organization supports inheritance of settings from the top to the bottom. With this powerful mechanism, a company policy can be created from a very strict setup (top profile) and weakens down the tree to the bottom profiles, representing your company organization structure. This Profile Tree can be accessed from menu *"Services / Profile Tree"*. An example of a Profile Tree can be seen in Figure 4.1, “Example profile tree”.

For example, an organization may define a strict global policy in the *organisation* profile. The child profiles, each representing various organizational units in the company, weaken this profile by allowing exceptions.

The *development* profile may allow access to specific sites with development resources and the *infrastructure* hive restrict access to all categories and applications, because the server infrastructure just needs web access to apply security updates.



- organisation
  - development
  - human resources
  - infrastructure
    - server center north
      - linux servers
      - windows servers
    - server center south
  - sales
    - group east
    - group west

● Allowed categories ○ Blocked categories

News
Finance
Travel
Real Estate
Vehicles
Education
Webmail
Shopping
Jobs
Communities
Search Engines/Directories
Resources

● Allowed applications ○ Blocked applications

● SSL intercepted categories ○ Not intercepted categories

Figure 4.1. Example profile tree

The screen shows the profile tree on the left side. When selecting a profile (left-click), the right part of the screen shows the effective policy of the most important filters as a quick overview.

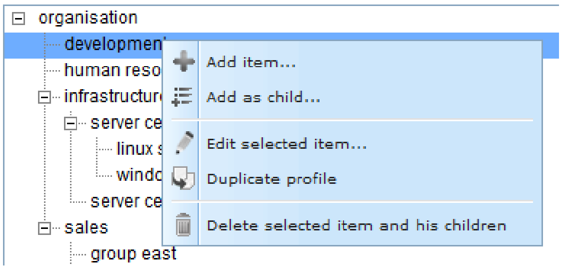


Figure 4.2. Profile tree context menu

A new Profile can be created by right-clicking on an existing profile and choosing *"Add item…"* to add a new profile on the same level as currently selected one, or *"Add as child…"* to create a child profile attached to the currently selected one. It is also possible to drag and drop profiles to move them in the hierarchy.

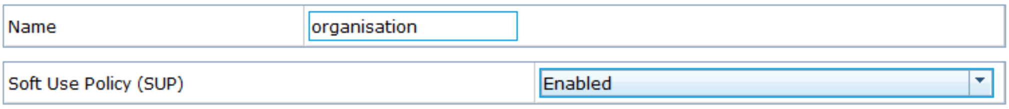
Profile settings for a specific profile can be opened by double-clicking a profile. A window opens with a detailed profile settings.

4.2. Profile Settings

The following section describes the profile settings and their effect on the traffic flowing through the Secure Web.

4.2.1. Setup

The profile setup screen lets the administrator change the profiles name and control the *Soft Use Policy*.



Name	organisation
Soft Use Policy (SUP)	Enabled

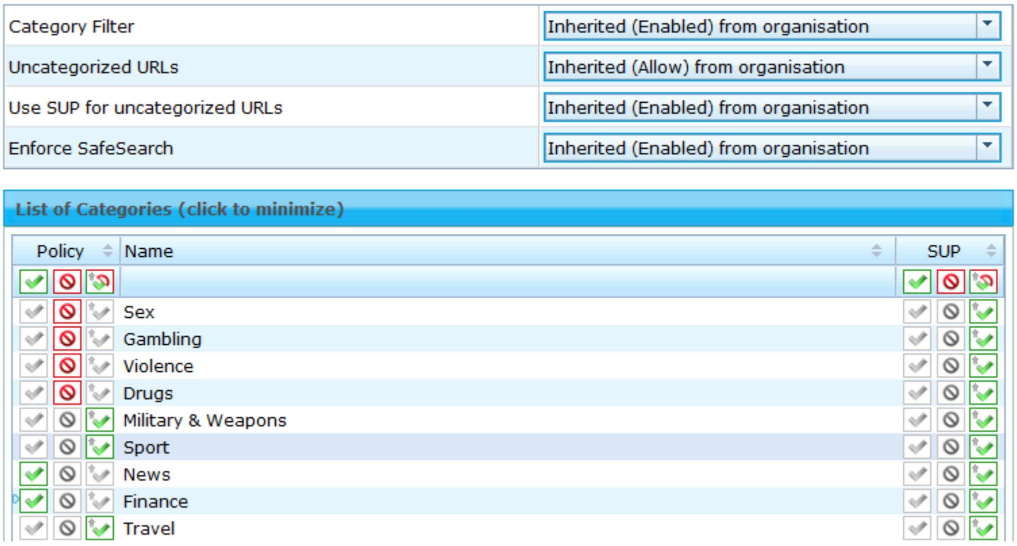
Figure 4.3. Profile setup

- **Name** - Current profile name, which may be changed here.
- **Soft Use Policy (SUP)** - Controls the profile wide availability of the *Soft Use Policy (SUP)* feature. If it is turned off here, *SUP* is not available in this profile. The settings may be inherited from parental profile.

Soft Use Poliy (SUP) allows a user of Secure Web to overrule filters that have been applied to his requests by himself. A blocking page then contains a button on which the user may click, overrule the blocking and continue surfing to the previously blocked content. The usage of *SUP* is logged and may also be alerted by e-mail.

4.2.2. Categories

Each requested URL that is handled by the Secure Web is checked with the CYAN URL database and classified into a set of 32 categories. A policy can be set up to control how the categories should be handled.



Category Filter	Inherited (Enabled) from organisation
Uncategorized URLs	Inherited (Allow) from organisation
Use SUP for uncategorized URLs	Inherited (Enabled) from organisation
Enforce SafeSearch	Inherited (Enabled) from organisation

List of Categories (click to minimize)

Policy	Name	SUP
✓ ⊘ ↺		✓ ⊘ ↺
⊘	Sex	↺
⊘	Gambling	↺
⊘	Violence	↺
⊘	Drugs	↺
↺	Military & Weapons	↺
↺	Sport	↺
✓	News	↺
✓	Finance	↺
↺	Travel	↺

Figure 4.4. List of categories

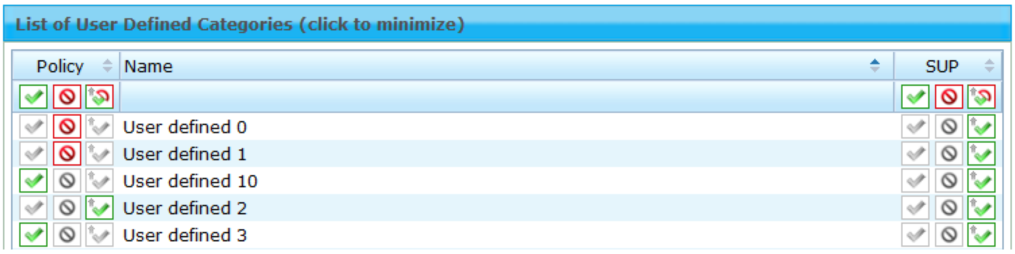
- **Category Filter** - Enables or disables the category filter in this profile. Please note that categorization is still done by the Secure Web for logging and reporting, even if the filter is turned off here.

- **Uncategorized URLs** - Controls the policy for URLs unknown to the URL database.

- **Use SUP for uncategorized URLs** - Controls if *Soft Use Policy* should be allowed for URLs blocked for being unknown to the URL database.

- **Enforce SafeSearch** - Enforces Safe Search techniques for web search engines like Google, Bing, Yahoo, and various other engines. If turned on, inappropiate content is automatically filtered by the search engines.

- **List of Categories** - A list of categories and their policy, including the policy of *SUP*, for each category. They can be set to *Allow*, *Deny* or *Inherit*. When creating a new profile the default value is *Inherit*.



List of User Defined Categories (click to minimize)

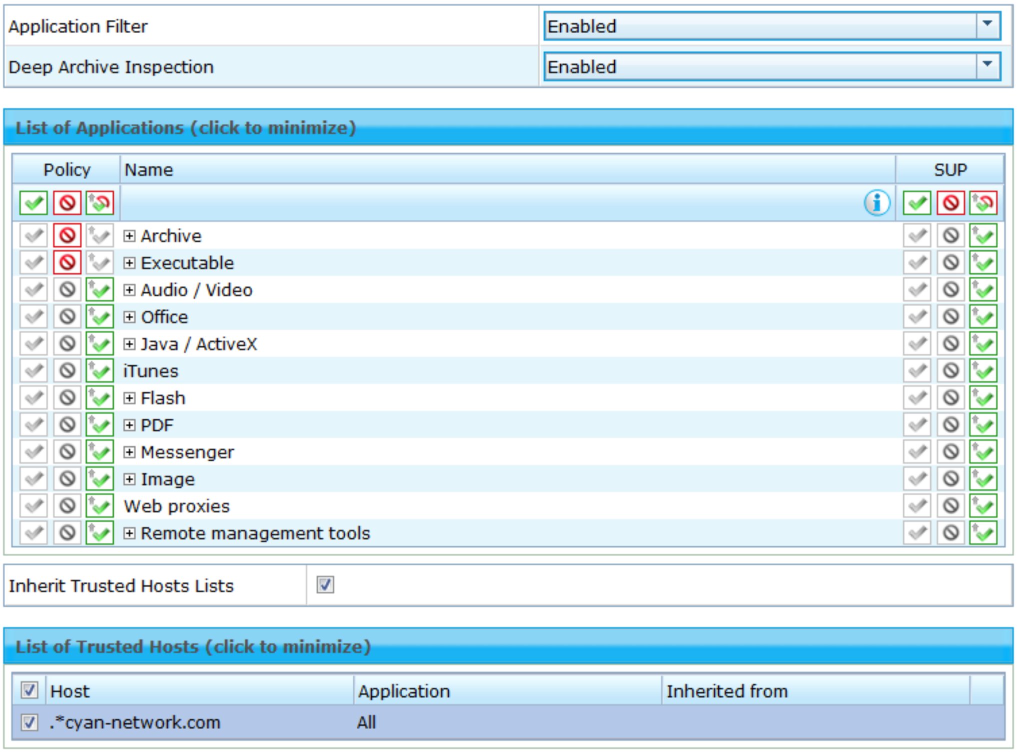
Policy	Name	SUP
✓ ⊘ ↺		✓ ⊘ ↺
⊘	User defined 0	↺
⊘	User defined 1	↺
✓	User defined 10	↺
↺	User defined 2	↺
✓	User defined 3	↺

Figure 4.5. List of user defined categories

- **List of User Defined Categories** - A list of user defined categories and their policy, including the policy of *SUP* for each category. The user defined categories can be edited by right clicking the list and selecting *"Edit User Defined Categories"* (you will be redirected to a different screen).

4.2.3. Applications

The application filter module combines different techniques to detect user applications, protocols and behaviors and allows the Secure Web to apply the company profile to the traffic.



Application Filter	Enabled
Deep Archive Inspection	Enabled

List of Applications (click to minimize)

Policy	Name	SUP
✓ ⊘ ↻	ⓘ	✓ ⊘ ↻
⊘	⊞ Archive	↻
⊘	⊞ Executable	↻
↻	⊞ Audio / Video	↻
↻	⊞ Office	↻
↻	⊞ Java / ActiveX	↻
↻	iTunes	↻
↻	⊞ Flash	↻
↻	⊞ PDF	↻
↻	⊞ Messenger	↻
↻	⊞ Image	↻
↻	Web proxies	↻
↻	⊞ Remote management tools	↻

Inherit Trusted Hosts Lists	☑

List of Trusted Hosts (click to minimize)

☑ Host	Application	Inherited from
☑ .*cyan-network.com	All	

Figure 4.6. Profile settings - Applications

- **Application Filter** - Enables or disables the application filter in this profile. Please note that application identification is always done by the Secure Web proxy engine for logging and reporting purposes, even if the filter is turned off here.

- **Deep archive inspection** - Enables the application filter to look inside archives for matching content and also enables the *Mime Types* filter to be applied for files in an archive.

- **List of Applications** - A list of application groups and their policy, including the policy of *SUP* for each application group. Groups can be expanded and policies defined for individual applications by clicking on the "+" symbol.

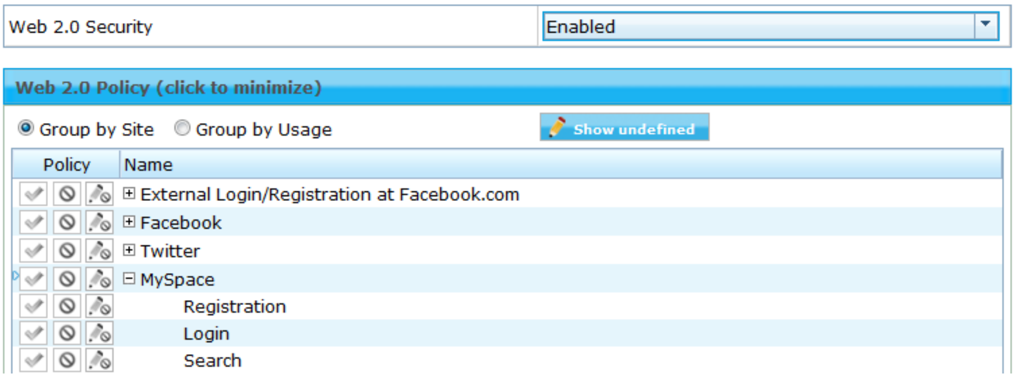
- **Inherit Trusted Hosts Lists** controls if the list of trusted hosts below should be inherited from the parent profile. If current profile does not have any parent profile (is in top level of the profile hierarchy), this option will not be present.

- **List of Trusted Hosts** contains a list of hosts that should be trusted. The policy for an application group will not be applied for hosts in this list.

4.2.4. Web 2.0

Web 2.0 security provides a way to deeply control the usage of typical Web 2.0 applications, such as Facebook, Twitter, etc. It allows the administrator to control access to these sites and, for every site, restrict usage of certain features like posting comments, chatting or uploading/download media content.

Any control of Web 2.0 sites here overrides the Category filter settings. Thus, if an application is explicitly allowed here, the Category policy for this site will have no effect. On the other hand if a policy is not defined here it can be controlled by a Category filter.



Web 2.0 Security: Enabled

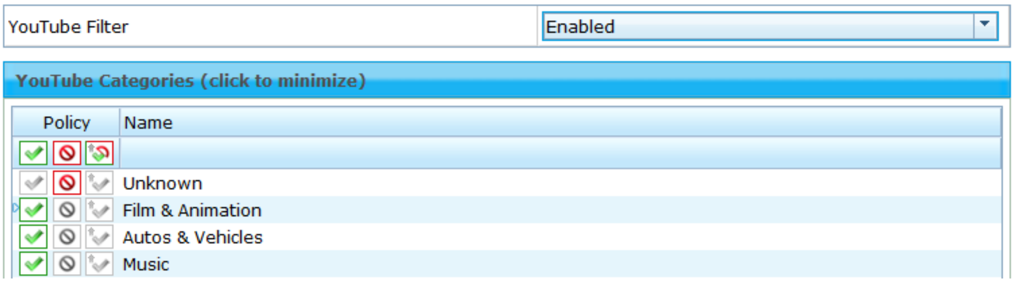
Web 2.0 Policy (click to minimize)

● Group by Site ○ Group by Usage

Policy	Name
✓ ⊘ ✎	⊞ External Login/Registration at Facebook.com
✓ ⊘ ✎	⊞ Facebook
✓ ⊘ ✎	⊞ Twitter
✓ ⊘ ✎	⊟ MySpace
✓ ⊘ ✎	Registration
✓ ⊘ ✎	Login
✓ ⊘ ✎	Search

Figure 4.7. Profile settings - Web 2.0

- **Web 2.0 Security** - Enables or disabled the Web 2.0 security module in this profile.
- **Web 2.0 Policy** - A list of Web 2.0 applications and sites specific usage. A policy can be defined either by site (*Group by Site* option) or by usage group (*Group by Usage* option, for example all Logins, Messaging, etc.).
- **Show undefined** button - Shows all options from the *Web 2.0 Policy* list that do not have any policy defined.



YouTube Filter: Enabled

YouTube Categories (click to minimize)

Policy	Name
✓ ⊘ ✎	
✓ ⊘ ✓	Unknown
✓ ⊘ ✓	Film & Animation
✓ ⊘ ✓	Autos & Vehicles
✓ ⊘ ✓	Music

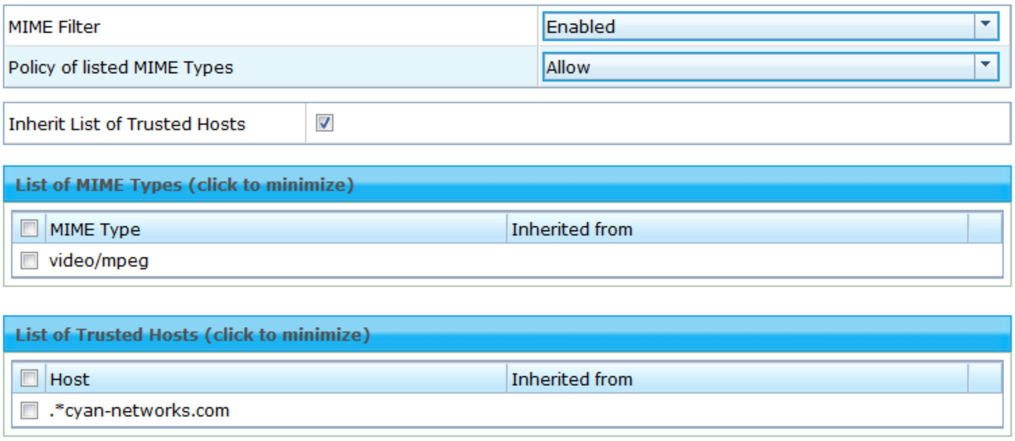
Figure 4.8. Profile settings - Youtube

- **YouTube Filter** - Enables or disables categorization of YouTube videos.
- **YouTube Categories** - A list of categories defined by Google for content on YouTube. Video links are identified by Secure Web and meta data checked against the category policy.

> Please note that categorization of YouTube content is not verified by CYAN Networks.

4.2.5. Mime-Types

The MIME filter is a basic filtering mechanism that is based on the MIME information set by a web server in the response to a request. Besides that, the Secure Web detects numerous content types itself, not trusting the given information from the untrusted web server.



MIME Filter	Enabled
Policy of listed MIME Types	Allow

Inherit List of Trusted Hosts	☑

List of MIME Types (click to minimize)

□ MIME Type	Inherited from
□ video/mpeg	

List of Trusted Hosts (click to minimize)

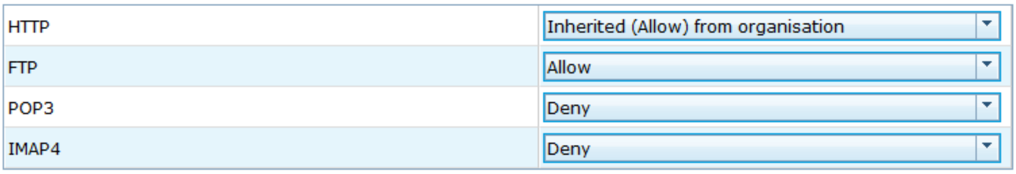
□ Host	Inherited from
□ .*cyan-networks.com	

Figure 4.9. Profile settings - MIME types

- **MIME Filter** - Enables or disables the MIME filter in this profile. Please note that MIME identification is always done by the Secure Web for logging and reporting, even if the filter is turned off here.
- **Policy of listed MIME Types** - Controls if the list of MIME types below should be seen as a list of allowed or denied MIME types, thus a white list or blacklist. If set to *Allow* the listed MIME types are the only ones allowed to pass through the Secure Web in this profile. If set to *Deny*, all of the listed MIME types will be blocked.
- **Inherit List of Trusted Hosts** - Controls if the list of trusted hosts below should be inherited from the parent profile.
- **List of MIME Types** - Contains a list of MIME types. New items can be added from a context menu displayed after right clicking the list.
- **List of Trusted Hosts** - Contains a list of hosts that should be trusted. The policy for MIME filtering will not be applied for hosts in this list.

4.2.6. Protocols

The Secure Web suite not only contains a classic HTTP(S) proxy, but also brings support for the FTP, IMAP4 and POP3 protocols in a native proxy engine.



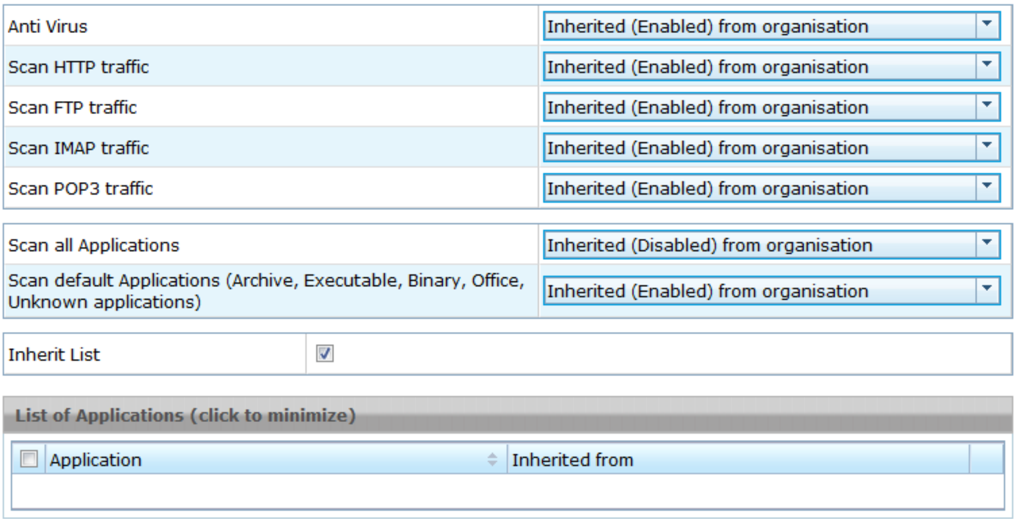
HTTP	Inherited (Allow) from organisation
FTP	Allow
POP3	Deny
IMAP4	Deny

Figure 4.10. Profile settings - Protocols

- **HTTP** - Enables or disables the HTTP(S) proxy service in the Secure Web Proxy.
- **FTP** - Enables or disables the FTP proxy service in the Secure Web Proxy.
- **POP3** - Enables or disables the POP3(S) proxy service in the Secure Mail Proxy.
- **IMAP4** - Enables or disables the IMAP4(S) proxy service in the Secure Mail Proxy.

4.2.7. Virus Scanning

Anti virus scanning is performend in all Secure Web proxy engines by the Secure Virus Scan Service and the infected downloads are denied for download.



Anti Virus	Inherited (Enabled) from organisation
Scan HTTP traffic	Inherited (Enabled) from organisation
Scan FTP traffic	Inherited (Enabled) from organisation
Scan IMAP traffic	Inherited (Enabled) from organisation
Scan POP3 traffic	Inherited (Enabled) from organisation
Scan all Applications	Inherited (Disabled) from organisation
Scan default Applications (Archive, Executable, Binary, Office, Unknown applications)	Inherited (Enabled) from organisation
Inherit List	☑

List of Applications (click to minimize)

□ Application	Inherited from

Figure 4.11. Profile virus scanning

- **Anti Virus** - Enables or disables the Virus Scan Service for this profile.

- **Scan HTTP traffic** - Controls if traffic going through the HTTP proxy engine should be scanned for virus.

- **Scan FTP traffic** - Controls if traffic going through the FTP proxy engine should be scanned for virus.

- **Scan IMAP traffic** - Controls if traffic going through the IMAP proxy engine should be scanned for virus.

- **Scan POP3 traffic** - Controls if traffic going through the POP3 proxy engine should be scanned for virus.

- **Scan all Applications** - If enabled, scans all data served through the Secure Web for viruses. Please note that scanning everything can have a considerable impact on the performance.

- Scan default Applications (Archive, Executable, Binary, Office, Unknown applications)* - If enabled, configures a default set of data types that should be scanned for virus. If disabled, the *List of Applications* below defines the application types that will be scanned for viruses.

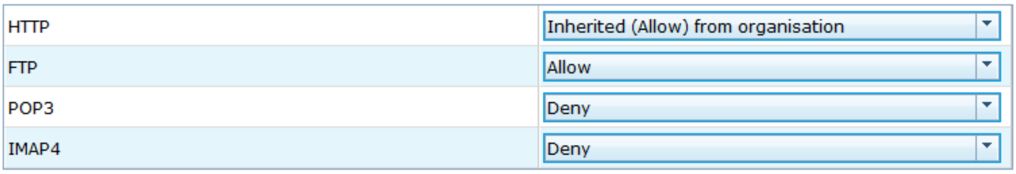
- **Inherit List** - Controls if the list of applications below should be inherited from the parent profile.

- **List of Applications** - Contains a list of application groups that will be scanned for viruses. To enable this list the options *Scan all Applications* and *Scan default Applications* must be set to *Disabled*.

> Anti Virus scanning needs the set up of the Virus Scan Service and a separate license to work correctly. Please see the documentation for Anti Virus Scanning for more information before enabling this filter.

4.2.8. SSL Tunneling

The Secure Web Proxy can also be used to tunnel arbitrary content. The SSL Tunneling filter controls if such content should be allowed.



HTTP	Inherited (Allow) from organisation
FTP	Allow
POP3	Deny
IMAP4	Deny

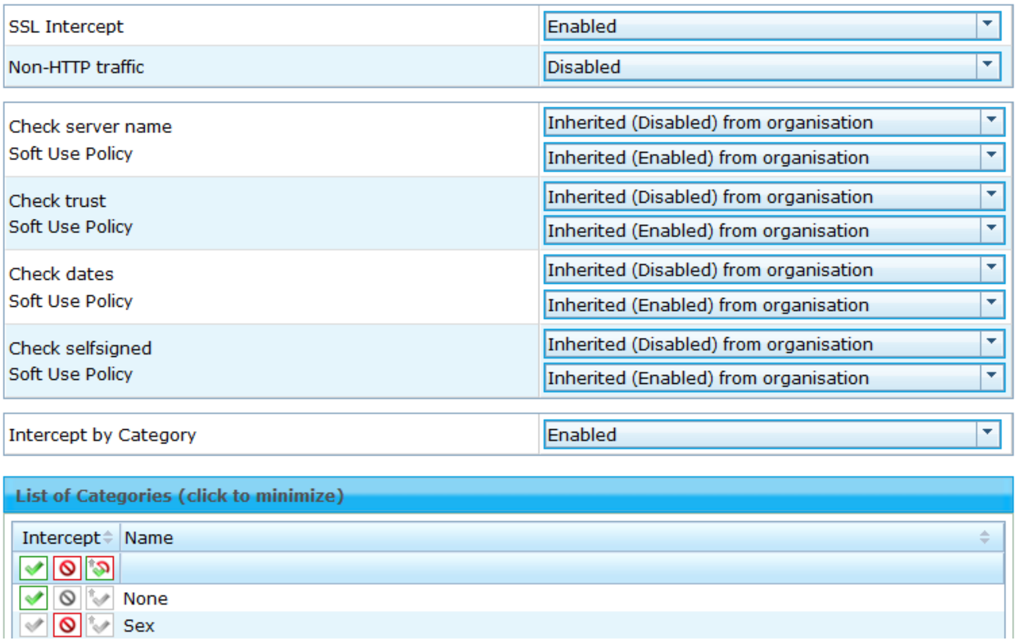
Figure 4.12. Profile settings - SSL Tunneling

- **Tunneling of non-SSL traffic** - Controls if non-HTTPS/SSL traffic is allowed to pass the HTTPS proxy engine. If disabled, checks are being made that only legit SSL traffic is allowed to pass the engine.
- **Inherit List** - Controls if the list of trusted hosts below should be inherited from the parent profile.
- **List of Trusted Hosts** - A list of hosts that should be trusted. The policy for SSL Tunneling filter will not be applied for hosts in this list.

> Certain protocols mimic SSL to be able to pass HTTPS proxies. One of these products is Skype, which tries to masquerade as SSL. Secure Web deploys strict checking of SSL and makes sure that only legit SSL traffic is allowed.

4.2.9. SSL Intercept

The SSL interception filter engine is capable of intercepting SSL traffic on the Secure Web. Traffic will be decrypted on the client side; the company policy is being applied and then re-encrypted for transmission to the server.



SSL Intercept	Enabled
Non-HTTP traffic	Disabled
Check server name	Inherited (Disabled) from organisation
Soft Use Policy	Inherited (Enabled) from organisation
Check trust	Inherited (Disabled) from organisation
Soft Use Policy	Inherited (Enabled) from organisation
Check dates	Inherited (Disabled) from organisation
Soft Use Policy	Inherited (Enabled) from organisation
Check selfsigned	Inherited (Disabled) from organisation
Soft Use Policy	Inherited (Enabled) from organisation
Intercept by Category	Enabled

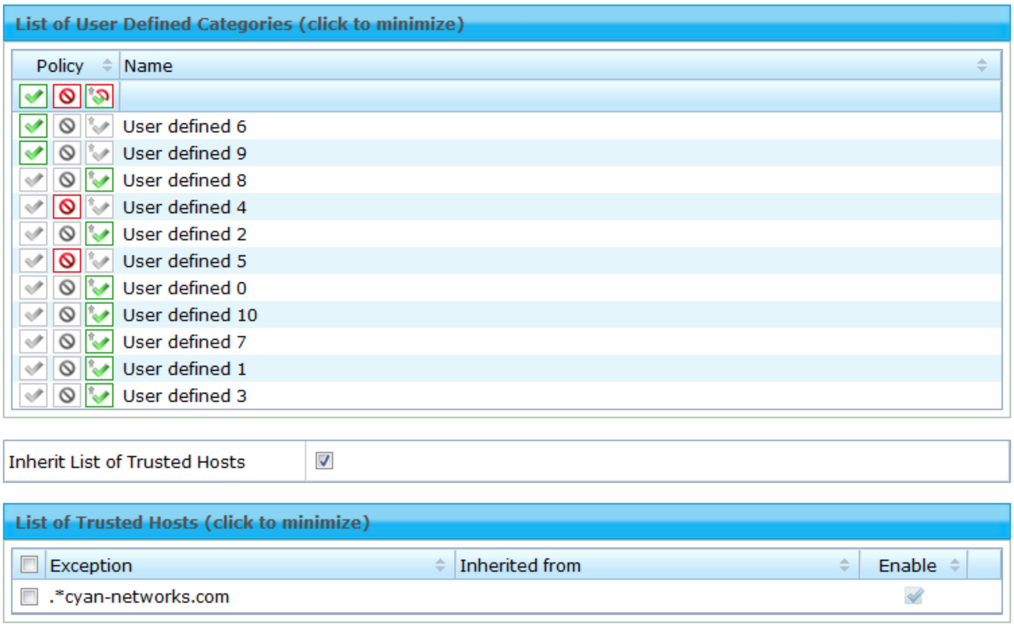
List of Categories (click to minimize)

Intercept	Name
✓ ⊘ ↺	
✓	None
⊘	Sex

Figure 4.13. Profile settings - SSL, part 1

- **SSL Intercept** - Enables or disables the SSL intercept engine for this profile.

- **Non-HTTP traffic** - Controls if non-HTTP traffic inside of SSL encrypted data streams should be allowed to pass the engine or not. Some protocols encapsulate their data inside SSL streams to hide from content inspections.
- **Check server name** - Controls if the server name of a server certificate should be checked against the request host name. If host names differ, the certificate will be considered invalid and a security violation page will be shown.
- **Check trust** - Controls if the server certificate chain should be checked against the Secure Web CA chain. If the chain can't be verified completely, the certificate will be considered invalid and a security violation page will be shown.
- **Check dates** - Controls if the server certificate expiration dates should be checked. If the expiration date has been reached or exceeded, the certificate will be considered invalid and a security violation page will be shown.
- **Check selfsigned** - Controls if self-signed certificates should be detected and defines a policy for them. If self-signed certificates are denied, the certificate will be considered invalid and a security violation page will be shown.
- **Soft Use Policy** - Controls if an end user is allowed to override the policy for the security alerts himself.
- **Intercept by category** - Controls if the decision to intercept SSL traffic should be based on a category information gained through URL categorization.
- **List of Categories** - Contains a list of categories for which SSL interception should be applied. This list is enabled just when *Intercept by Category* is enabled.



List of User Defined Categories (click to minimize)

Policy	Name
✓ ⊘ ↑✓	
✓	User defined 6
✓	User defined 9
↑✓	User defined 8
⊘	User defined 4
↑✓	User defined 2
⊘	User defined 5
↑✓	User defined 0
↑✓	User defined 10
↑✓	User defined 7
↑✓	User defined 1
↑✓	User defined 3

Inherit List of Trusted Hosts ☑

List of Trusted Hosts (click to minimize)

☐	Exception	Inherited from	Enable
☐	.*cyan-networks.com		✓

Figure 4.14. Profile settings - SSL, part 2

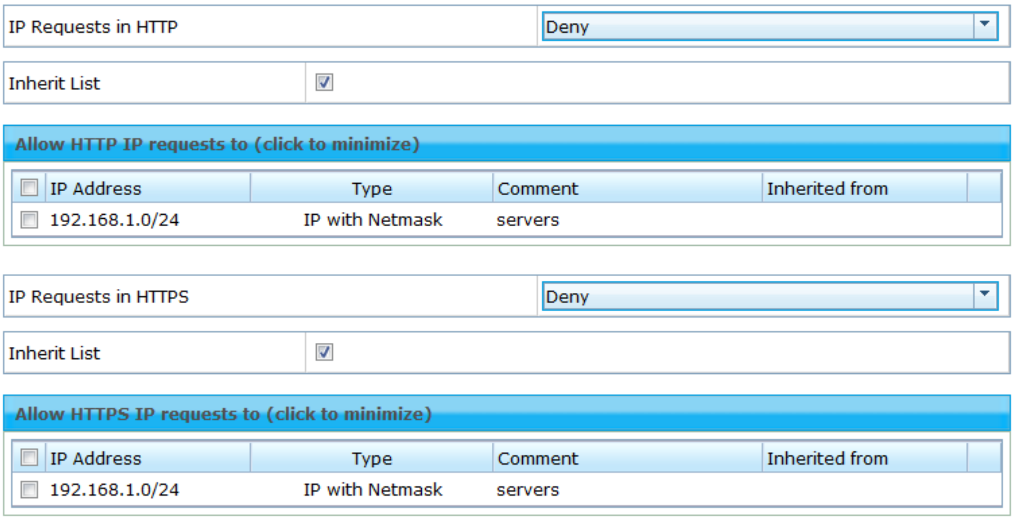
- **List of User Defined Categories** - A list of user defined categories for which SSL interception should be applied. Categories in this list can be modified by right clicking the list and selecting appropriate menu item.
- **Inherit List of Trusted Hosts** - Controls if the list of trusted hosts below should be inherited from a parent profile.

- **List of Trusted Hosts** - A list of hosts that should be trusted. SSL interception will not be done for hosts in this list.

SSL Interception needs the set up and rollout of a Certification Authority (CA) in the company network to work correctly. Please see the documentation for SSL Interception for more information before enabling this filter.

4.2.10. IP Requests

Requests containing URL in form of an IP address are uncommon and often used for phishing attacks. As such, the IP Request filter can prevent requests to URLs containing only IP addresses.



IP Requests in HTTP: Deny

Inherit List ☑

Allow HTTP IP requests to (click to minimize)

□ IP Address	Type	Comment	Inherited from
□ 192.168.1.0/24	IP with Netmask	servers	

IP Requests in HTTPS: Deny

Inherit List ☑

Allow HTTPS IP requests to (click to minimize)

□ IP Address	Type	Comment	Inherited from
□ 192.168.1.0/24	IP with Netmask	servers	

Figure 4.15. Profile settings - IP Requests

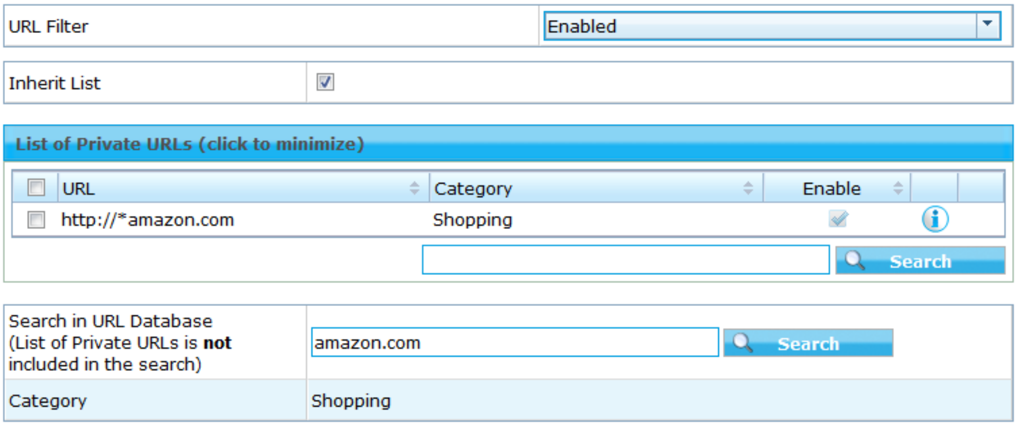
- **IP Requests in HTTP** - Controls if IP requests of HTTP requests should be filtered.
- **Inherit List** - Controls if the list of trusted hosts below should be inherited from the parent profile.
- **Allow HTTP IP requests to** - A list of IP addresses to which IP requests are allowed.
- **IP Requests in HTTPS** - Controls if IP request of HTTPS requests should be filtered.
- **Inherit List** - Controls if the list of trusted hosts below should be inherited from the parent profile.
- **Allow HTTPS IP requests to** - Contains a list of IP addresses to which IP requests are allowed.

Even though requests with IP addresses are uncommon in the Internet nowadays, they are still common in internal corporate environment. The list of IP addresses for both HTTP and HTTPS sites should contain the internal IP ranges to avoid any problem with internal services.

4.2.11. URL Filter

The built-in URL Filter allows filtering requests based on Category information of hosts from either CYAN URL Database or IBM Content Security. The URL Filter, or Secure Categorization Service, lets the administrator create his own category information or override existing categorization with new values.

Entries in the Private URL list always have precedence over built-in categorization.



URL Filter	Enabled
Inherit List	☑

List of Private URLs (click to minimize)

□	URL	Category	Enable	
□	http://*amazon.com	Shopping	✓	ⓘ

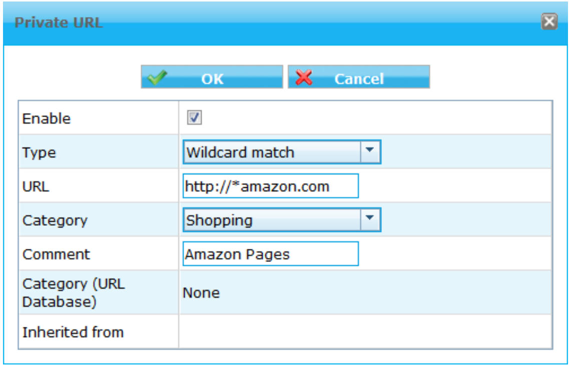
Search in URL Database (List of Private URLs is **not** included in the search)	amazon.com
Category	Shopping

Figure 4.16. Profile settings - URL Filter

- **URL Filter** - Enables or disables the private URL list for this profile.
- **Inherit List** - Controls if the list of private URL entries below should be inherited from the parent profile.
- **List of Private URLs** - A list of URL patterns and category assignments. All assignments done here have precedence over standard URL categorization of CYAN URL Database or IBM Content Security SDK.

There is also present a search box for searching in the current URL Database for reference or checking if for some URL a category already exists.

If the URL Filter feature is enabled, you can add new entries to the *List of Private URLs* by right clicking the list and selecting *Add item…*.



Private URL

Enable	☑
Type	Wildcard match
URL	http://*amazon.com
Category	Shopping
Comment	Amazon Pages
Category (URL Database)	None
Inherited from	

Figure 4.17. Profile settings - Adding new Private URL

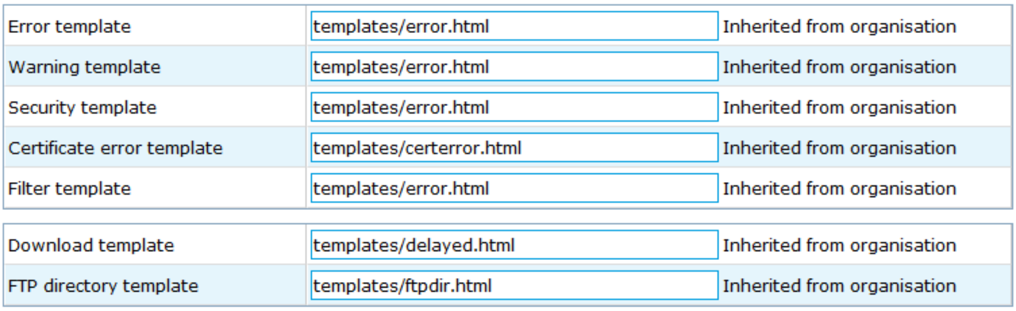
- **Enabled** - Indicates if the entry is currently enabled and thus matching will be performed.
- **Type** - The matching type. Supported are these matching types:
  - **Wildcard match** - Expects an URL with a maximum of one wildcard (`'*'`) at either the beginning **or** end of hostname in the *URL* entry (e.g. http://*.amazon.com/shop).
  - **Regular Expression** - Expects a regular expression in *URL* and matches the requested URL against the regular expression (e.g. http://.*\.amazon\..*/).
  - **Full match** - Tries to fully match the request URL against the *URL* (e.g. http://www.amazon.com/).

- **URL** - Contains the URL, or parts of it, that should be matched.
- **Category** - Sets a new category for matching URL. This may either be one of the supplied categories from the built-in URL Database, a user defined category or *None* if no category information should be applied.
- **Comment** - Contains user-specific comments for the entry, for example the reason why an entry has been made.
- **Category (URL Database)** - Prints out the Category information of the supplied *URL* from the built-in URL Database for reference (if exists).
- **Inherited from** - Contains the name of the profile which defines this policy entry.

Private URL List allows different matching algorithms applied to each entry.

4.2.12. Templates

Templates let the administrator define HTML templates for certain pages used to inform the user of specific situations on a per profile level. Sample templates are provided with the Secure Web installation and may be modified to match specific needs (company CI/CD, Helpdesk information, etc).



Error template	templates/error.html	Inherited from organisation
Warning template	templates/error.html	Inherited from organisation
Security template	templates/error.html	Inherited from organisation
Certificate error template	templates/certerror.html	Inherited from organisation
Filter template	templates/error.html	Inherited from organisation

Download template	templates/delayed.html	Inherited from organisation
FTP directory template	templates/ftpdir.html	Inherited from organisation

Figure 4.18. Profile settings - Templates

- **Error template** - A template used to notify the user of an error (invalid HTTP request, networking problems, etc).
- **Warning template** - A template used to warn the user of possible problems (e.g. server certificate can’t be read).
- **Security template** - A template used to notify the user of a violation of the security policy (e.g. SSL certificate failed to verify).
- **Certificate error template** - A template used to notify the user of certificate problems during SSL intercept (e.g. server name mismatch, certificate has expired, etc).
- **Filter template** - A template used to notify the user of a violation of the filter policy (e.g. URL categorization, IP requests, Application blocking, etc).
- **Download template** - A template used when buffering of data is applied for the sake of Anti Virus scanning.
- **FTP directory template** - A template used to render a directory listing when accessing FTP resources in the FTP-over-HTTP proxy engine.

The default example files are located in directory */opt/cyan/sweb/templates* on the physical file system of the Appliance.

5. Profiles Assignment Primer

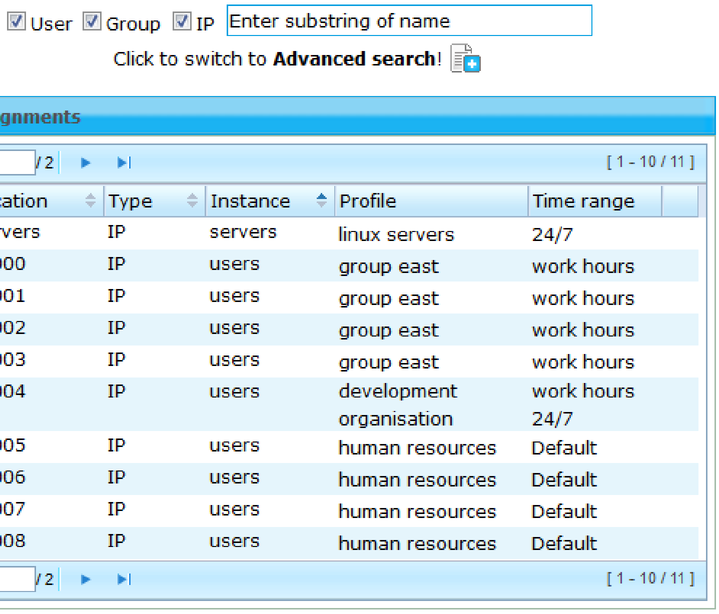
Profile assignment is the place where all information come together to form the actual policy for a request that is processed through the Secure Web.

Based on the available client authentication information, be it a username, a group name or an IP address, the available credentials are mapped to an actual profile. If no mapping can be found, the default profile will be used.

No request can pass the proxy engine without a profile being applied.

5.1. Assignments

Assigning a profile to an identification of an IP address, a user or a group can be done in menu *Services / Profile Assignment / Profile Assignment*. The Profile Assignment dialog gives you an overview over the current assignments that are being applied on the Secure Web.



☑ User ☑ Group ☑ IP Enter substring of name

Click to switch to **Advanced search**!

Profile Assignments

1 /2 [1 - 10 / 11]

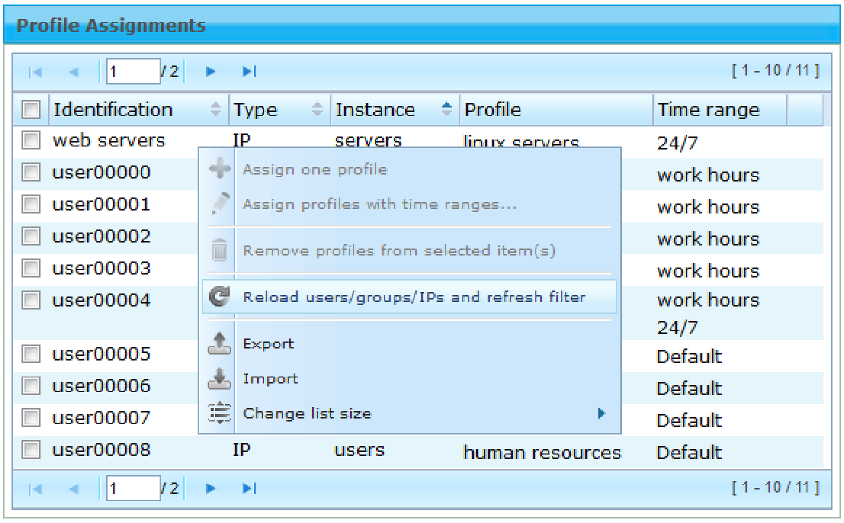
□	Identification	Type	Instance	Profile	Time range
□	web servers	IP	servers	linux servers	24/7
□	user00000	IP	users	group east	work hours
□	user00001	IP	users	group east	work hours
□	user00002	IP	users	group east	work hours
□	user00003	IP	users	group east	work hours
□	user00004	IP	users	development organisation	work hours 24/7
□	user00005	IP	users	human resources	Default
□	user00006	IP	users	human resources	Default
□	user00007	IP	users	human resources	Default
□	user00008	IP	users	human resources	Default

1 /2 [1 - 10 / 11]

Figure 5.1. Profile assignments overview

Depending on the chosen search filter, the screen shows a list of assignments, each consisting of the credential name, the type of this credential (IP, Group or User), the authentication instance this credential is coming from, the selected profile and the time range profile.

5.1.1. Fetching the list of credentials



Profile Assignments
1 / 2
[1 - 10 / 11]
Assign one profile
Assign profiles with time ranges...
Remove profiles from selected item(s)
Reload users/groups/IPs and refresh filter
Export
Import
Change list size
1 / 2
[1 - 10 / 11]

□ Identification	Type	Instance	Profile	Time range
□ web servers	IP	servers	linux servers	24/7
□ user00000				work hours
□ user00001				work hours
□ user00002				work hours
□ user00003				work hours
□ user00004				work hours 24/7
□ user00005				Default
□ user00006				Default
□ user00007				Default
□ user00008	IP	users	human resources	Default

Figure 5.2. Setting up Profile Assignment, context menu

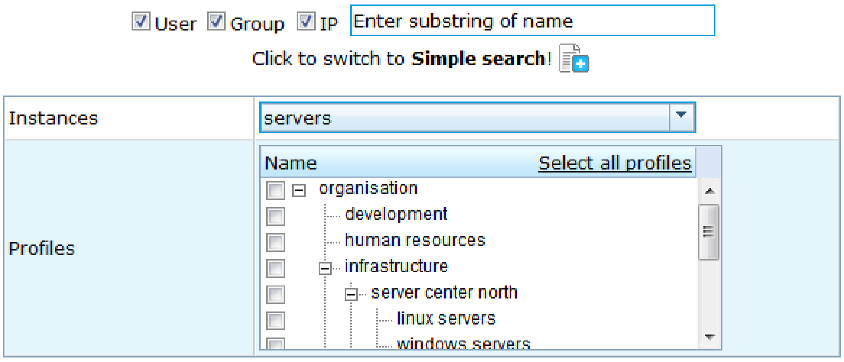
If the list of users and/or groups has changed on the authentication instance, the list has to be reloaded manually. To do this, select the *Reload users/groups/IPs and refresh filter* right-click context menu item.

The list of user credentials will be fetched and display filters applied.

5.1.2. Setting up display filters

To customize the information an administrator will see on the Profile Assignment dialog, a display filter can be set.

The simple search filters that you could see in Figure 5.1, “Profile assignments overview”, lets the administrator quickly filter the list of users and groups by type (User, Group or IP) as well as parts of the object name.



☑ User ☑ Group ☑ IP Enter substring of name

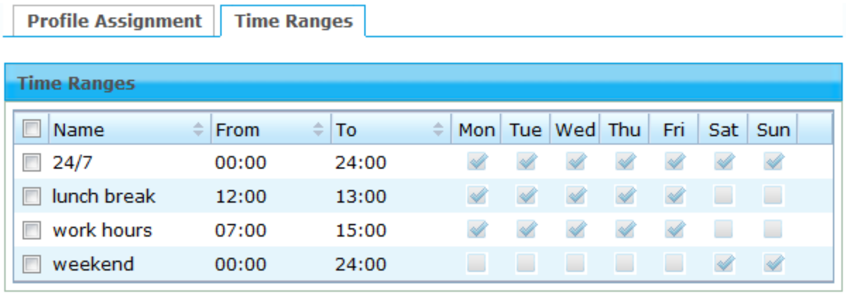
Click to switch to **Simple search**!

Instances	servers
Profiles	Name — Select all profiles □ ⊟ organisation □ development □ human resources □ ⊟ infrastructure □ ⊟ server center north □ linux servers □ windows servers

Figure 5.3. Setting up Profile Assignment filter (Advanced)

By clicking on *Advanced search*, the administrator may now also search in specific authentication instances and for objects that have specific profiles assigned.

5.1.3. Defining Time Profiles



□	Name	From	To	Mon	Tue	Wed	Thu	Fri	Sat	Sun
□	24/7	00:00	24:00	✓	✓	✓	✓	✓	✓	✓
□	lunch break	12:00	13:00	✓	✓	✓	✓	✓	□	□
□	work hours	07:00	15:00	✓	✓	✓	✓	✓	□	□
□	weekend	00:00	24:00	□	□	□	□	□	✓	✓

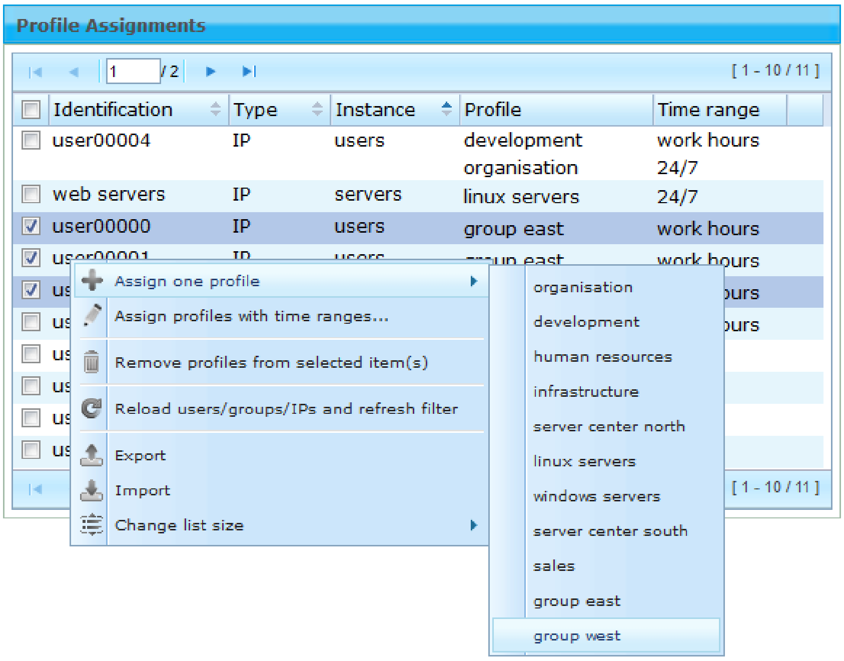
Figure 5.4. Setting up Time Ranges

Time Ranges can be defined in the *Services / Profile Assignment / Time Ranges* tab. The administrator may add, edit or delete time ranges that may later be used to restrict profile assignments to certain days of a week and a time range. New time range can be added from the context menu.

A Time Profile *Default* always exists and cannot be deleted. This profile spans all weekdays and has a time range of 00:00 to 24:00.

5.1.4. Assigning a profile

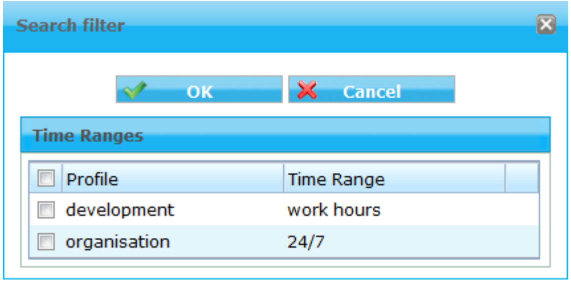
In tab *Services / Profile Assignment / Profile Assignment*, a profile can be assigned to one or more user credentials by checking their checkboxes and choosing the desired profile in the right-click context menu (as you can see in the following figure).



□	Identification	Type	Instance	Profile	Time range
□	user00004	IP	users	development organisation	work hours 24/7
□	web servers	IP	servers	linux servers	24/7
☑	user00000	IP	users	group east	work hours
☑	user00001	IP	users	group east	work hours
☑	us				ours
□	us				ours
□	us				
□	us				
□	us				
□	us				

Figure 5.5. Profile assignment

Per default, a profile assignment is set to the *Default* Time Profile mentioned in the previous section. By setting a different Time Profile, an assignment can be limited to certain days and time span of a week. To assign a Time Profile, select a user credential and choose *Assign profiles with time ranges…* from the context menu (also visible in Figure 5.5, “Profile assignment”).



□	Profile	Time Range
□	development	work hours
□	organisation	24/7

Figure 5.6. Assigning a Time Profile

In the new window, you can add to the list entries that connect a profile to a specific time range by right clicking the empty list and selecting *Add item* option. Existing entries in the list can be modified by double clicking them and changing the values.

The order of entries is important here, as time ranges may overlap. The first one that matches the time range, when the profile assignment is evaluated is taken into effect. Entries may be reordered by drag & drop of the list elements.

5.1.5. Deleting an assignment

Assignments can be deleted by selecting one or multiple entries from the Profile Assignment list, right clicking any of them and selecting *Remove profiles from selected item(s)* from the context menu (as could see in Figure 5.5, “Profile assignment”).

Appendix A. Contact data

A.1. How to contact our sales department

Tel.:	+43 (1) 33933-0
Email:	sales@cyan-networks.com

A.2. How to contact our support department

Tel.:	+43 (1) 33933-333
Email:	support@cyan-networks.com

A.2.1. Getting Support

In case you should have any technical problems, or questions and would like to get support from our team, we kindly ask you to provide us with the following information:

- Description of your question or problem
- The version information of the product:
  - The version information of Secure Web can be found after logging into the Web Admin Interface in the top part of the screen:

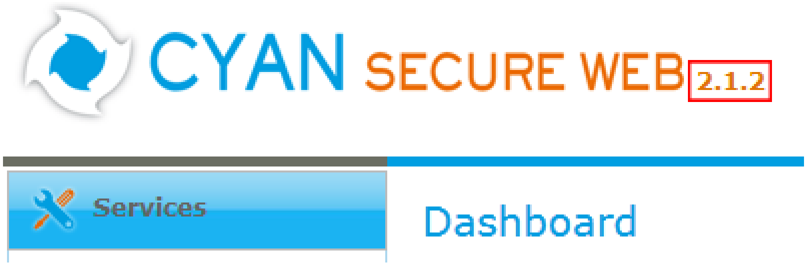


Figure A.1. Version information of the Secure Web

  - The version information of the Reporting System can be found after login in the top part of the screen of the Web Admin Interface:

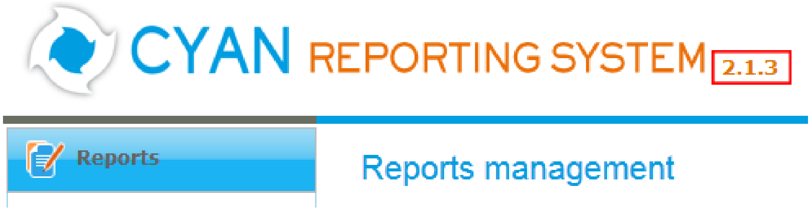


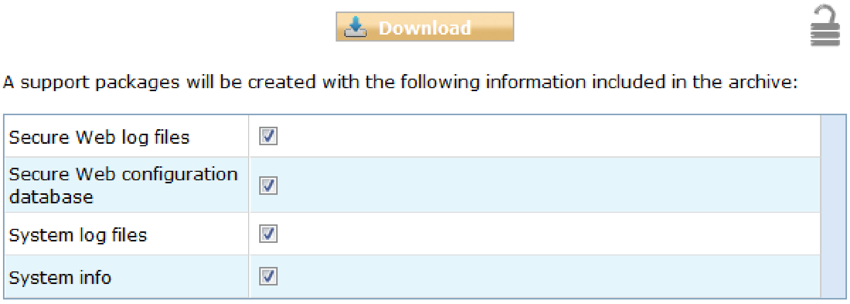
Figure A.2. Version information of the Reporting System

  - All the information contained in the screen found in menu *Services / Services / Overview*
- In the case authentication is activated, provide us with the method in place (via Windows Agent, via Linux Agent, etc.)
- The deployment method of the Appliance (Out-of-line, In-Line, DMZ)
- The operation mode of the Appliance (dedicated mode, transparent mode)

- Information about the environment (proxy cascades that are used, firewalls and gateways involved in the infrastructure that are of relevance to the Appliance)

The appliance interface provides the possibility to create a support package that includes the configuration and log files of the system. This package can help us to track down the issue easier and faster. Please attach this package to your e-mail.

In order to create a support pack, navigate to menu *Appliances / Maintenance / Support* and click on the *Download* button. You may select the files you want to provide to our support team and then download a package, which we kindly ask you to send to our support email address.



A support packages will be created with the following information included in the archive:

Secure Web log files	☑
Secure Web configuration database	☑
System log files	☑
System info	☑

Figure A.3. Support Package

Further documentation about the product as well as technical white papers that describe certain use cases can be found in our documentation repository on our homepage:

http://www.cyan-networks.com/documentation