# Allowing or Denying Client Certificates

https://campus.barracuda.com/doc/4259860/

The **ACCESS CONTROL > Client Certificates** page allows you to define allow/deny rules based on client certificates. These settings are not used unless **Enable Client Authentication** is *Yes* for the Service on the **BASIC > Services** page. For more information on service settings, see Step 3: Configuring Basic Service Settings.

When Client Authentication is turned on for a service, all clients are required to present a certificate to access the website. The certificate is first checked for validity. A valid certificate cannot have expired, and must be signed by a certificate authority (CA) listed under Trusted Certificates for the service. Even a valid certificate signed by a trusted CA can be rejected based on the certificate attributes. This is useful when you wish to revoke an issued valid certificate.

## How It Works

Each Allow/Deny rule has the following important attributes:

- A sequence number specifying the order in which to evaluate the rule.
- A set of attribute matches (like certificate serial number). The attribute can either be a wildcard match (*, to indicate match any value), or it can be a specific value, matching the certificate's corresponding attribute exactly.
- An action to take when the presented client certificate matches this rule.

When a request is received, the client certificate is compared to all Allow/Deny rules in sequence number order, starting from the lowest sequence number. Each attribute in the rule is compared, and if all attributes match a rule, the corresponding action (Allow or Deny) is taken and no further rules are compared.

When no rule matches the client certificate in the request, the request is allowed by default.

To allow only requests whose client certificates match a rule, create a Deny rule with a high sequence number (255, for example), which matches all rules (has * for all attributes) and the action **Deny**. Every request with a client certificate that fails to match a rule will be denied. Each allowed certificate must have a corresponding Allow rule with a lower sequence number.

> If you create a high sequence number Deny rule to deny all except explicitly allowed certificates, a request will be allowed only if its certificate and all certificates in its chain match an Allow rule. If its intermediate or trusted certificate does not match any rule, the request will be denied.

Complex rules can be built using Allow/Deny rules. For example, to deny all certificates from the Sales Department except one that is identified by its serial number, create the following two rules:

- Sequence = 1; Action = Allow; Organizational Unit = Sales; Serial Number = 12345
- Sequence = 2; Action = Deny; Organizational Unit = Sales

While complex rules can be built if needed, the recommended configuration allows all certificates signed by a trusted CA and uses the Allow/Deny list only to revoke access for issued certificates that are no longer valid. The certificate serial number can uniquely identify a certificate issued by a single CA in the event that it must be revoked. The common name can also be used to identify a revoked certificate.

#### Configuring Allow/Deny Certificate Rules

Detailed instructions for configuring Allow/Deny certificate rules are available on the **ACCESS CONTROL > Client Certificates** page by clicking **Help** on that page. When providing values for the sequence, ensure that you provide a number between 1 and 255. This is to indicate the order of processing when matching rules. Rules with a lower number are matched first, and attempts to match will stop at the first rule that matches. A rule with sequence number 1 will be matched first, and a rule with sequence number 255 will be matched last.

> In order for a certificate to be allowed via an Allow rule, ensure that Allow rules also exist for all certificates in its chain. If the certificate itself matches an Allow rule, but its intermediate or trusted certificate does not match any rule, the request will be denied.