Autoconfig: How to create a configuration file

Definition

Authoritative definition

Example

Real-world example

```
<?xml version="1.0" encoding="UTF-8"?>

<clientConfig version="1.1">
  <emailProvider id="freenet.de">
    <domain>freenet.de</domain>
    <displayName>Freenet Mail</displayName>
    <displayShortName>Freenet</displayShortName>
    <incomingServer type="imap">
      <hostname>imap.freenet.de</hostname>
      <port>993</port>
      <socketType>SSL</socketType>
      <authentication>password-encrypted</authentication>
      <username>%EMAILADDRESS%</username>
    </incomingServer>
    <incomingServer type="imap">
      <hostname>imap.freenet.de</hostname>
      <port>143</port>
      <socketType>STARTTLS</socketType>
      <authentication>password-encrypted</authentication>
      <username>%EMAILADDRESS%</username>
    </incomingServer>
    <incomingServer type="pop3">
      <hostname>pop.freenet.de</hostname>
      <port>995</port>
      <socketType>SSL</socketType>
      <authentication>password-cleartext</authentication>
      <username>%EMAILADDRESS%</username>
    </incomingServer>
    <incomingServer type="pop3">
      <hostname>pop.freenet.de</hostname>
      <port>110</port>
      <socketType>STARTTLS</socketType>
      <authentication>password-cleartext</authentication>
      <username>%EMAILADDRESS%</username>
    </incomingServer>
    <outgoingServer type="smtp">
      <hostname>smtp.freenet.de</hostname>
      <port>465</port>
      <socketType>SSL</socketType>
      <authentication>password-encrypted</authentication>
      <username>%EMAILADDRESS%</username>
    </outgoingServer>
    <outgoingServer type="smtp">
      <hostname>smtp.freenet.de</hostname>
      <port>587</port>
      <socketType>STARTTLS</socketType>
      <authentication>password-encrypted</authentication>
      <username>%EMAILADDRESS%</username>
    </outgoingServer>
    <documentation url="http://kundenservice.freenet.de/hilfe/email/programme/config
      <descr lang="de">Allgemeine Beschreibung der Einstellungen</descr>
      <descr lang="en">Generic settings page</descr>
    </documentation>
    <documentation url="http://kundenservice.freenet.de/hilfe/email/programme/config
      <descr lang="de">TB 2.0 IMAP-Einstellungen</descr>
      <descr lang="en">TB 2.0 IMAP settings</descr>
    </documentation>
  </emailProvider>
</clientConfig>
```

How to probe mail servers

To determine a server's capabilities, you can contact the server directly and talk the POP/IMAP/SMTP protocol manually (assuming you already know the hostname).
For non-SSL, use **`netcat -v`** ***`hostname port`*** (preferred) or `telnet` *`hostname port`* as "client".

- POP3, port 110: when you see `+OK WEB.DE POP3-Server` or similar, enter **`CAPA`**, hit return.
- IMAP, port 143: when you see `* OK mwinf2j04 IMAP4 server ready` or similar, enter **`1`**

`CAPABILITY`, hit return.

- SMTP, port 587 or 25: when you see `220 mail.gmx.net GMX Mailservices ESMTP` or similar, enter **`EHLO example.net`**, hit return.

In all cases, the server should respond with a list of capabilities.

SSL / STARTTLS

There are 2 SSL variants: normal SSL and STARTTLS.

Normal SSL

The old-style SSL (including TLS, which is just the new name for SSL) has a special port:

- POP3 via SSL: port 995
- IMAP via SSL: port 993
- SMTP via SSL: port 465

On Linux, you can contact the server via

```
openssl s_client -connect hostname:port
```

You should see output about the SSL handshake and the certificate. Important is what is listed as "CN=". This must be the same as the hostname that you contacted, otherwise the certificate is not valid (or you need to use another hostname).
If you see nothing, then probably the server does not support SSL.
After that, you can have the same protocol exchange as with netcat on standard ports, as listed above.

STARTTLS

STARTTLS is a special, new form of SSL, which works on the standard ports (e.g. port 143 for IMAP). You can contact the server via netcat as mentioned above. If you see "STARTTLS" (for IMAP, SMTP) or "STLS" (for POP) listed as one of the capabilities, the server should support STARTTLS.
To try it out, on Linux, you can contact the server via

```
openssl s_client -connect hostname:port -starttls proto
```

...where "proto" is `imap`, `pop3` or `smtp`. For example:

```
openssl s_client -connect imap.example.com:143 -starttls imap
```

You should get the same response as described above for openssl.

Configuration file format

Add the appropriate port and socket type for each server, depending on protocol and SSL support. For example,
for IMAP with SSL:

```
<port>993<port>

<socketType>SSL</socketType>
```

for IMAP with STARTTLS:

```
<port>143<port>
<socketType>STARTTLS</socketType>
```

for IMAP without any SSL (deprecated!):

```
<port>143<port>

<socketType>plain</socketType>
```

Use SSL

Please do not submit or serve any configurations without SSL! There's no reason in 2010 why users still need to read mail entirely unprotected.
If you are an ISP and think the server load is too high, try adding an SSL accelerator. They are cheap and widely used. In fact, even most freemail (!) providers these days support SSL, so if users pay you money for ISP service, that's all the more reason to give them proper service. But first simply try to enable software SSL - small servers may be fine with SSL and without any additional installations.

Valid certificate

Either way, be sure to use a valid certificate:

- issued by a CA recognized by Firefox / Thunderbird
- not expired
- the CN in the cert must match the hostname that Thunderbird contacts and that is listed as <hostname> in the configuration file. If they don't match, Thunderbird must assume that the user may be under attack, otherwise the SSL guarantees no longer hold. Thunderbird either warns the user or ignores the server. (Note that you can get certificates for free these days, for example from startssl.com.)

Authentication

Probe the mail server, as explained above for STARTTLS. If you see `CRAM-MD5` or `APOP` in the response, the server should support encrypted passwords. If you *only* see `AUTH LOGIN` and/or `PLAIN`, or no `AUTH` at all, the server probably does not support secure authentication. In the former case, select "Encrypted passwords" as "Authentication method" (in Thunderbird Account Settings UI, incoming server and SMTP server), and test whether you can actually log in with a real account (because some servers are unfortunately broken with regards to authentication, often due to a wrong or misconfigured SASL installation).

Configuration file format

In the configuration file, for each IMAP, POP and SMTP server, you need to specify the authentication method.

For plaintext passwords:

```
<authentication>password-cleartext</authentication>
```

For CRAM-MD5:

```
<authentication>password-encrypted</authentication>
```

Discouraged settings (SMTP only):
If the SMTP server can only be used after checking incoming mail, please use

```
<authentication>smtp-after-pop</authentication>
```

Note that RFC 4409 disallows that and requires the customer-facing SMTP server to support proper authentication via AUTH.
If the SMTP server can only be used within the ISP's network, and requires no authentication, use:

```
<authentication>client-IP-address</authentication>
```

or, if it requires authentication in addition to the user being in the ISP network, use e.g.:

```
<authentication>password-cleartext</authentication>

<restriction>client-IP-address</restriction>
```

However, that means that users on the road or in the office are unable to send mail, which is a real problem for many of our users. This violates RFC 4409 as well and is an outdated configuration. Please try find a configuration that works in all cases, for the sake of the users.

Please support MD5 passwords

Please support authentication with CRAM-MD5. It is simple to implement, and to set up, and you can still use RADIUS or a database that stores passwords in plaintext, so you don't need to make changes to your mail server or authentication infrastructure apart from installing some software and configuring it correctly. CRAM-MD5 is particularly important when no SSL is used: Never make users send their passwords in plaintext over the network! (Not even in your ISP network.) We warn users in the Mail Account Creation dialog about such insecure configurations, and we reserve the right to block them in the future.

As an ISP, you should ideally store passwords in encrypted format, which removes the risk of mass password theft (and possibly reuse on other sites) if somebody hacks your servers. You can still support plaintext passwords in this case, and encrypt passwords on the fly before comparing. (Users who use plaintext passwords would still be somewhat exposed, but at least you don't have the risk of the whole plaintext password database being stolen.) You can use both plaintext and encrypted authentication transmission with plaintext or encrypted password databases - the two issues are independent.

Username

If the user's IMAP login name is the same as his email address (for example, if "fred@example.com" is the login name), add:

```
<username>%EMAILADDRESS%</username>
```

Note: Use %EMAILADDRESS% as literal. Thunderbird will replace it with the email address that the user entered. Same for %EMAILLOCALPART% and other placeholders.

If the login name is the same as the first segment before the @ of the email address (for example, "fred" for "fred@example.com"), use:

```
<username>%EMAILLOCALPART%</username>
```

Aliases, or username not part of email address

Note that the above must be true for any email address that the user would set up - even for aliases.

You can ignore aliases like info@, if that's an alias for fred@ (or both fred@ and wilma@) and Fred would set up fred@example.com in Thunderbird, not info@.

If, however, Fred can set up hero@example.com as alias for fred@example.com, and neither "hero" nor "hero@example.com" would work as login name on your IMAP server, you need to set up a lookup of alias -> username on your autoconfig server. So, if you get a request for <http://autoconfig.example.com/mail/c...ro@example.com>, your autconfig server must have a script which responds to /mail/config-v1.1.xml and returns the concrete username, for example:

```
<username>fred</username>
```

... (or <username>fred@example.com</username>, as appropriate) for hero@example.com. This is the only way to enable automatic configuration without users having to remember what their primary login name was, which is a serious problem in practical experience. Even if you have told them all the necessary information in your welcome letter, they usually cannot find the letter. That's exactly where autoconfiguration tries to help.

Enable visiturl

Some providers do not provide IMAP or POP service by default, but require it to be enabled via a web

UI. If that is the case, add the URL that a logged-in user would use into this field, and the application can prompt the user to visit it.

This is not yet supported by Thunderbird 3.1, but should be in the future, so please add this critical information where it applies.

If you are an ISP, please by all means avoid this. It's one of those "walls" against which users run the hard way.

Documentation URL

If the configuration is (partially) based on a help webpage of the ISP that describes the configuration that end users should use, you can record its URL here. You may add several of them, as several elements. It is for informational purposes only and mainly for the maintenance of the config file, the client currently does not use them at all.

If your URL contains ampersands (`&`), please remember to replace them with HTML entities (`&amp;`). For example:

```
<documentation url="http://example.com/help.php?client=thunderbird&amp;lang=en"/>
```

Otherwise your XML file will be incorrect and Thunderbird will neither be able to parse it, nor to return any error message.