Information Note
Vega Provisioning

Vega gateways can be configured through their Web Browser and Command Line interfaces but where mass deployment and consistency of configuration is required the Vega autoexec and scheduled autoexec functionality provide an excellent way of configuring Vega gateways. If an on demand update is required a SIP Notify can be sent to the Vega which, after authentication, will trigger the Vega to load a configuration.

At power on, and from Release 8.2 at timed intervals or on demand, Vega gateways can be configured to collect a command file from a tftp, ftp, http or https server, and then act upon the commands contained therein.

Typically the commands in the command file will instruct the Vega to check whether it is running the same firmware as that loaded on the server, and instruct the Vega to check whether it is running with the latest configuration. If either needs updating the Vega will load the relevant firmware / configuration and start using it.

The following paragraphs explain a typical provisioning process: initial configuration followed by regular update. The concepts explained here can be used to implement other schemes as required.

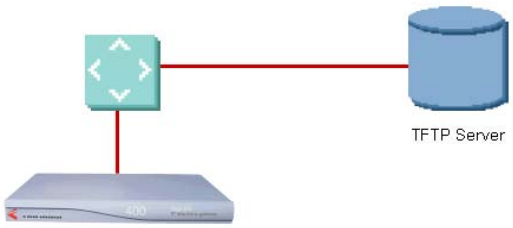
Overview of an initial configuration followed by regularly updated configuration

To configure a Vega gateway from a central configuration server:

1. Connect the new Vega to a LAN segment local to the initial configuration tftp server – this configures the Vega with a basic configuration that will provide it with https information and the required update period so that once it is deployed it can use https to collect further updates.

2. Connect the Vega at its permanent location – on powering up the Vega, it will immediately attempt a configuration update and then following that it will retry for updates at the time interval specified in the tftp loaded configuration, or, if a subsequent https update has modified the time period, at the time period most recently configured into the Vega.

Initial (TFTP) configuration



When a Vega is connected and turned on for the first time it uses DHCP to pick up local IP information, including its own IP address and the IP address of a tftp server.

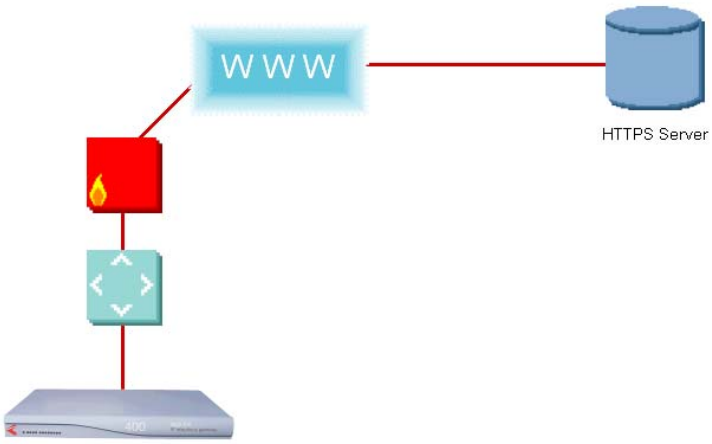
Once the Vega has its IP information, the Vega autoexec functionality requests the tftp server to supply a command file `<MAC_address>script.txt`[1]

If the tftp server cannot supply this file then the Vega requests the file `defaultscript.txt`

Whether it is best to use the MAC address based filename, or the `defaultscript.txt` filename will depend on whether all gateways require the same initial configuration, or whether different gateways require different configurations.

If all can use `defaultscript.txt`, only a single file is required to be served by the tftp server. If the MAC address is to be used then one configuration file will be needed for each Vega configured (or a program driving an instrumented tftp server will be required to serve the required text configuration file based on the MAC address contained in the filename requested).

Update configuration



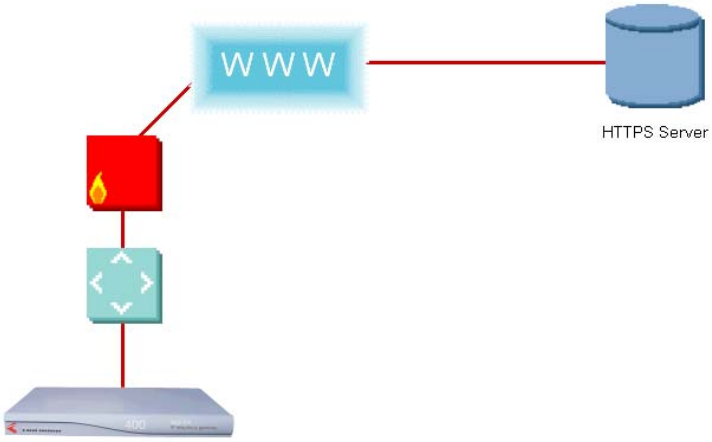
When the Vega is powered up on site it will immediately use the autoexec configured settings to try and pick up a new configuration. The tftp loaded configuration will have configured this to access an https server. The tftp configuration will also have had the opportunity to alter the filename that the Vega tries to collect. This may remain `defaultscript.txt` if all Vegas are to be kept identically configured, or left configured to pick up `<MAC_address>script.txt`, alternatively if

[1] Where <MAC_address> is the MAC address of the first LAN interface of the Vega with no separator between octets, e.g. 005058123456script.txt

groups of Vegas need to be configured the same as others in their group but differently from other groups then the script file name may be updated to a group name.

After the Vega is up and running it will use the scheduled autoexec function in the Vega to request further updates at timed intervals. Typically the command file that scheduled autoexec requests will be exactly the same command file as the autoexec function requests – because both processes aim to bring the Vega up to date with the latest configuration.

On demand configuration



If a new configuration needs to be loaded, the Vega can be triggered to immediately initiate a config load by sending it a SIP NOTIFY message.

The sender of the NOTIFY message will have to pass an authentication challenge before the config load will be actioned.

The filename of the command file and the method of accessing the command file are specified in the Content-Type header of the Notify message.

e.g.

```
Content-Type: message/external-body; access-type="URL"; URL="http:/Steve/VegaStream/005058040070_notify.txt"
```

See appendix 1 for an example trace of a NOTIFY requesting a config load.

Why use tftp for the initial configuration then use https?

tftp

tftp is a simple protocol that works well in local networks. It is UDP based – so there are no link layer retries, and it has no encryption, but as it is only used locally in a safe environment this is no problem.

The main reason for using tftp for the initial configuration is that DHCP servers can supply a tftp server IP address as part of their served information. (DHCP cannot supply ftp, http or https IP addresses). Using tftp the first configuration can be completely automatic, as soon as the Vega is connected and powered up, it will collect its IP address and the tftp server address information. It will then ask the tftp server for the command file to load.

https
The https protocol runs over TCP and therefore has link-layer resends built in which provides reliable end to end connection meaning that this protocol is much better suited for longer distance connections and connections where some packet loss may be encountered.

https also provides encryption of data across the link, and so avoids external parties being able to snoop upon and view the configuration information being transmitted between the server and the Vega.

https is also a protocol used as part of web browsing – it is therefore one of the protocols that firewalls usually allow through by default (unlike ftp or tftp). Use of https therefore minimises the likelihood of firewall transmission problems.

What is contained in a configuration file for download to the Vega?

The command file will typically look like:

```
upgrade
download enable
download firmware VegaEuropa_R081S001.abs reboot ifnew
exit
get %mconfig.txt ifdiff save rebootifneededwhenidle apply
```

NOTE

There MUST be a blank line after the last command line in the autoexec script file as the Vega needs to see the Carriage Return at the end of the command line in order to execute the command.

For further details about Vega Autoexec command files see the Vega Primer on www.VegaAssist.com

The configuration file contains commands just as though they were typed on the command line interface of the Vega, e.g. to set the parameter _advanced.autoexec.scriptfile1 to group1_command_file.txt and lan.file_transfer_method to https the configuration file will contain the two lines:

```
set ._advanced.autoexec.scriptfile1=group1_command_file.txt
set .lan.file_transfer_method=https
```

For further details about parameters that can be configured in the Vega see the Vega Primer on www.VegaAssist.com

To act as a template, or to see the format of the data required in a configuration file, archive the Vega configuration using the CLI command:

```
put vega_cfg.txt
```

The Vega will send the file vega_cfg.txt to the server specified in the file_transfer_method parameter. The format of this file is exactly the format that the Vega expects to see in its autoexec configuration file. Not all parameters need to be defined so delete lines where you do not want to configure that parameter, and edit lines that need a parameter set to a specific value.

At the top of a configuration file there should be a line of the form:

```
; CONFIGVERSION:<name>:<date as DD/MM/YY> <time as HH:MM:SS>
```

e.g.

```
; CONFIGVERSION:Steve_Hight_Vega:28/11/2006 17:46:39
```

This is used by the Vega to decide whether the configuration file needs to be loaded (whether it is different from the previous configuration file loaded). If a configuration file is modified on the server, ensure that the header is modified (e.g. to the date/time of the change) so that when a Vega checks to see whether the file has changed it will see that it is a new configuration and so will load it.

Configuring Autoexec

Autoexec has the following parameters:

```
[_advanced.autoexec]
    enable=1
    lastconfig=none
    scriptfile1=%iscript.txt
    scriptfile2=defultscript.txt
```

`enable`	enables and disables autoexec at boot time
`lastconfig`	should not be changed as it contains information from the header of the previously loaded config – to decide whether a new config file is different
`scriptfile1`	1st script file to look for and execute
`scriptfile2`	2nd script file to look for and execute (if script file 1 is not found)

Configuring Scheduled Autoexec

Scheduled Autoexec has the following parameters:

```
[cron.entry.n}
    enable=1
    script=blank
    when=never
```

`enable`	enables and disables scheduled autoexec
`script`	script file to collect and execute
`when`	times at which to execute the script file

when can take the values:

- `Never` = do not execute ever
- space separated values for “minute" "hour" "day of month" "month" "day of week”
  - where values can be:

*	matches every minute, hour etc.
n	one specific minute/hour/etc.
n,m	a comma-delimited list of matching minutes/hours/etc.
n-m	an inclusive range of minutes/hours/etc.
/n	"every n intervals" used to modify a range
sun,mon	day names, used in "weekday" column
jan,dec	month names, used in "month" column

e.g.

12 23-7/2 * * sat,sun
will run a script at 12 minutes past every other hour (because of the "/2") between 23 (11pm) and 7 (7am), on every Saturday or Sunday.

12 0-6 * 7 *
will run a script at 12 minutes past the hour between 0 (midnight) and 6 (6am), but only during the month of July.

Configuring NOTIFY authentication details

Notify authentication needs the following parameters configured:

```
[sip.remote_admin]
     realm=default_realm

[sip.remote_admin.1]
     enable=0
     password=default
     username=default
```

`realm`	the realm for the authentication
`enable`	enables and disables this remote admin user
`password`	authentication password (if set = 'default' then authentication will always fail)
`username`	authentication username (if set = 'default' then authentication will always fail)

Monitoring configuration load status

If SNMP is enabled, Vega sends out traps to indicate the success or otherwise of autoexec.

For further details about configuring SNMP in the Vega see the Vega Primer and the SNMP information note, both available on www.VegaAssist.com

Syslog and other Vega logging may also be watched to see the results.

To enable monitoring at the SIP messaging level, from release 8.2 the Vega allows the last part of the User agent ID in the SIP header to be modified (see parameter `_advanced.sip.user_agent_header_ext`)

This can be configured and updated as new configuration files are loaded onto the Vega.

Appendix 1 – Example NOTIFY message exchange to request a config load

```
SIP m:1480342 141002 00009<-- UA RX --- From  UDP(18):172.19.1.233:5060
  NOTIFY sip:service@172.19.1.230:5060 SIP/2.0
  Via: SIP/2.0/UDP 192.168.0.1:5060;branch=z9hG4bK-14823-1-0
  From: sipp <sip:sipp@192.168.0.1:5060>;tag=14823SIPpTag001
  To: sut <sip:service@172.19.1.230:5060>
  Call-ID: 1-14823@192.168.0.1
  CSeq: 1 NOTIFY
  Contact: sip:sipp@192.168.0.1:5060
  User-Agent: Provisioning
  Event: ua-profile
  Max-Forwards: 70
  MIME-Version: 1.0
  Content-Type: message/external-body; access-type="URL";
URL="http:/Steve/VegaStream/005058040070_notify.txt";
  Content-Length: 0

SIP m:1480347 0005 00010--- UA TX --> To     UDP(18):172.19.1.233:5060
  SIP/2.0 401 Unauthorized
  v: SIP/2.0/UDP 192.168.0.1:5060;branch=z9hG4bK-14823-1-0;received=172.19.1.233
  f: sipp <sip:sipp@192.168.0.1:5060>;tag=14823SIPpTag001
  t: sut <sip:service@172.19.1.230:5060>
  i: 1-14823@192.168.0.1
  CSeq: 1 NOTIFY
  m: <sip:service@172.19.1.230:5060>
  WWW-Authenticate: Digest realm="test_ntfy", qop="auth",
nonce="D39C218A8E2A733FD5727A59FA96789F16FC019DFE17E3D9"
  User-Agent: VEGAEURO/13.02.08.2xS003 got_http
  l: 0

SIP m:1480352 0005 00011<-- UA RX --- From  UDP(18):172.19.1.233:5060
  NOTIFY sip:service@172.19.1.230:5060 SIP/2.0
  Via: SIP/2.0/UDP 192.168.0.1:5060;branch=z9hG4bK-14823-1-2
  From: sipp <sip:sipp@192.168.0.1:5060>;tag=14823SIPpTag001
  To: sut <sip:service@172.19.1.230:5060>
  Call-ID: 1-14823@192.168.0.1
  CSeq: 1 NOTIFY
  Contact: sip:sipp@192.168.0.1:5060
  Authorization: Digest
username="c2",realm="test_ntfy",cnonce="6b8b4567",nc=00000001,qop=auth,uri="sip:172.19.1.230:5060",n
once="D3
9C218A8E2A733FD5727A59FA96789F16FC019DFE17E3D9",response="6515aa56a5d81067b3866f851d660de1",algorith
m=MD5
  User-Agent: Provisioning
  Event: ua-profile
  Max-Forwards: 70
  MIME-Version: 1.0
  Content-Type: message/external-body; access-type="URL";
URL="http:/Steve/VegaStream/005058040070_notify.txt";
  Content-Length: 0

SIP m:1480362 0010 00012--- UA TX --> To     UDP(18):172.19.1.233:5060
  SIP/2.0 200 OK
  v: SIP/2.0/UDP 192.168.0.1:5060;branch=z9hG4bK-14823-1-2;received=172.19.1.233
  f: sipp <sip:sipp@192.168.0.1:5060>;tag=14823SIPpTag001
  t: sut <sip:service@172.19.1.230:5060>
  i: 1-14823@192.168.0.1
  CSeq: 1 NOTIFY
  m: <sip:service@172.19.1.230:5060>
  User-Agent: VEGAEURO/13.02.08.2xS003 got_http
  l: 0
```


Contact Details
Email: support@vegastream.com
Web: www.vegastream.com
www.vegaassist.com

EMEA Office
VegaStream Limited
The Western Centre
Western Road
Bracknell
Berks RG12 1RW
UK

+44 (0) 1344 784900

USA Office
VegaStream Inc.
6200 Stoneridge Mall Road
3rd Floor
Pleasanton
California 94588
USA

+1 925 399 6428