MINIMAL
SDWN CORE
GETTING STARTED GUIDE

User Experience
Architecture Overview
Installation
Minimal Configuration
TR-069 Integration
Splunk Integration

Anyfi Networks
Västergatan 31 B
21121 Malmö
Sweden
info@anyfinetworks.com
Contents

Preface ........................................................................................................................................ v
Intended Audience ..................................................................................................................... v
Document Conventions .............................................................................................................. v
  Advisory Paragraphs .................................................................................................................. v
  Typographic Conventions ......................................................................................................... vi
Chapter 1: User Experience ........................................................................................................ 7
SIMPLE – Home Wi-Fi Anywhere .................................................................................................. 7
Chapter 2: Architecture Overview ............................................................................................. 8
SIMPLE – Home Wi-Fi Anywhere .................................................................................................. 8
Chapter 3: Installation .................................................................................................................. 10
System Requirements .................................................................................................................. 10
Automated Provisioning on VMware ESXi ................................................................................... 10
  Preparations .............................................................................................................................. 10
  Instructions ............................................................................................................................... 11
Automated Provisioning on VirtualBox ....................................................................................... 12
  Preparations .............................................................................................................................. 12
  Instructions ............................................................................................................................... 12
Manual Provisioning on VirtualBox ............................................................................................. 12
  Preparations .............................................................................................................................. 12
  Instructions ............................................................................................................................... 12
Completing the Installation ......................................................................................................... 14
Chapter 4: Configuration .............................................................................................................. 15
Configuring the Controller .......................................................................................................... 15
  Basic Networking ...................................................................................................................... 15
Preface

This document details how to install, configure and verify a minimal example SDWN core.

This example combines our Controller with a basic management server. The resulting core is capable of running our SIMPLE solution.

**NOTE** This example SDWN core is part of the Community Edition of our Carrier Wi-Fi System. Community Edition is unsupported and restricted to a maximum of 100 radios and services, but can be used for both commercial and non-commercial purposes. Contact sales@anyfinetworks.com for other licensing options.

Intended Audience

This document is intended for system and network administrators with previous experience of Command Line Interfaces (CLIs) for administrative tasks. Readers should have specific knowledge in the following areas:

- Networking and data communications
- Internet protocol (IP)

Readers lacking experience with the Vyatta Network OS are encouraged to study its online documentation.

Document Conventions

This guide contains advisory paragraphs and uses the below typographic conventions.

Advisory Paragraphs

This guide uses the following advisory paragraphs:

**Warnings** alert you to situations that may pose a threat to your system or subscriber's security, as in the following example:

⚠️ **WARNING** Bridging unauthenticated Wi-Fi traffic to a network interface may pose a security threat to the associated network.

**Cautions** alert you to situations that might affect service, as in the following example:
CAUTION Restarting a production system will interrupt service.

Notes provide important information about the structure or functioning of the system:

NOTE The Controller is a controller in the Software-Defined Networking (SDN) sense of the word, not in the typical corporate WLAN sense.

Typographic Conventions

This document uses the following typographic conventions:

<table>
<thead>
<tr>
<th>Typographic Conventions</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monospace</td>
<td>Examples, command-line output, and representations of configuration nodes. Also commands, keywords, and file names, when mentioned inline.</td>
</tr>
<tr>
<td><strong>bold Monospace</strong></td>
<td>Your input: something you type at a command line.</td>
</tr>
<tr>
<td><em>italics</em></td>
<td>An argument or variable where you supply a value. Also concepts and principles when mentioned inline.</td>
</tr>
<tr>
<td><code>&lt;key&gt;</code></td>
<td>A key on your keyboard, such as <code>&lt;Enter&gt;</code>. Combinations of keys are joined by plus signs (“+”), as in <code>&lt;Ctrl&gt;+c</code>.</td>
</tr>
<tr>
<td>`[ arg1</td>
<td>arg2 ]`</td>
</tr>
<tr>
<td><code>num1–numN</code></td>
<td>A inclusive range of numbers. An example is <code>1–65535</code>, which means 1 through 65535, inclusive.</td>
</tr>
<tr>
<td><code>arg1..argN</code></td>
<td>A range of enumerated values. An example is <code>eth0..eth3</code>, which means <code>eth0</code>, <code>eth1</code>, <code>eth2</code>, or <code>eth3</code>.</td>
</tr>
<tr>
<td><code>arg[ arg...]</code></td>
<td>A value that can optionally represent a list of elements (a space-separated list in the first case and a comma-separated list in the second case).</td>
</tr>
</tbody>
</table>
Chapter 1: User Experience

In this chapter we describe the end-user experience that can be achieved with this minimal example SDWN core.

SIMPLE - Home Wi-Fi Anywhere

Our SIMPLE solution provides every user with seamless and secure remote access to their home Wi-Fi network. We call this a “zero sign-on” user experience because there is absolutely nothing the end-user needs to do to connect. There is no separate registration process, no software that needs to be installed on the device, not even any settings to change – it just works.

Note that each user will see their own home Wi-Fi SSID – but not each other’s. This ensures that everybody can seamlessly connect to their network, without cluttering the list of networks in the connection manager.

Advanced radio policy control allows the operator to set a minimum quality of service level that must be reached before the network is even presented to the mobile device. This helps avoid the most common user experience problem in carrier Wi-Fi deployments: that mobile devices connect too early to the network – when the radio link quality is so poor that meaningful communication on Layer 3 is not possible.

The user data plane is protected end-to-end, all the way from the mobile device to the user's own home gateway, by the well-known and trusted WPA2 security mechanism. Not even an attacker in complete control of the access point can eavesdrop on or modify a guest’s communication.

Seamless hand-over from CPE to CPE with session continuity on Layer 3 is fully supported.

For more information about the SIMPLE solution and its user experience please see http://www.anyfinetworks.com/solutions/simple.
Chapter 2: Architecture Overview

This example SDWN core contains a management machine and a Controller. The management machine is not part of our product portfolio, and serves only as an example and a placeholder for a third party management system.

![Network diagram of the minimal example SDWN core.](image)

**Figure 1**: Network diagram of the minimal example SDWN core.

Software-Defined Wireless Networking (SDWN) is an SDN overlay architecture, with the control plane centralized in a Controller while the data plane remains distributed. Both user payload and control traffic are transported over UDP/IP, requiring no specific network integration. Access points can therefore be installed anywhere there is IP connectivity.

As you can see in Figure 1 you will need an IP address to install and run this minimal SDWN core. In this guide we assume that this is a static public IP address and we refer to it as *ip1*.

**SIMPLE - Home Wi-Fi Anywhere**

Fixed-line subscribers are provided seamless and secure remote access to their home Wi-Fi networks by the SIMPLE SDWN App. In this minimal example the user payload is routed “peer-to-peer”, encapsulated in a UDP/IP tunnel, from the visited access point to the home gateway – and from there to the Internet. This means that users retain their home IP address at all times, ensuring traceability, lawful intercept and seamless handovers between access points.
**NOTE** SDWN data plane tunnels carry the raw encrypted wireless protocol frames inside UDP/IP datagrams. In this system the air interface is IEEE 802.11 and the inner tunnel protocol is thus IEEE 802.11 Mac Protocol Data Units (MPDUs).

The CPE and the Controller are the only two integration points for this basic implementatoin of the SIMPLE solution. This architecture may however not be ideal for access technologies like ADSL where the uplink bandwidth is significantly lower than the downlink, or where there is a significant cost associated with upstream traffic (e.g. DOCSIS). For these types of networks we recommend incorporating an Optimizer for central breakout of Internet-bound traffic.
Chapter 3: Installation

In this chapter we describe how to install the minimal example SDWN core on VMware ESXi or Oracle VM VirtualBox.

System Requirements

Your host system should meet the following requirements:

- 64-bit x86 CPU with Intel VT-x or AMD-V instruction set
- VMware ESXi 5.x hypervisor or host OS supported by Oracle VM VirtualBox:
  - Linux 2.6 or 3.x
  - Windows 64-bit Vista, Server 2008, 7, 8, Server 2012 or Server 2013
  - Mac OS X 10.6, 10.7, 10.8 or 10.9
- Minimum of 1 GB RAM available for VMs
- Minimum of 15 GB free disk space for VM disk images
- At least one Ethernet network interface

Automated Provisioning on VMware ESXi

In this section we explain how to use the self-extracting provisioning script to provision a minimal example core on a host system running the VMware ESXi 5.x hypervisor.

Preparations

If you have not yet installed the VMware ESXi hypervisor on the host system you will of course need to do so before we begin. An evaluation version of ESXi 5.5, also known as vSphere Hypervisor 5.5, is freely available and can be downloaded from VMware's website.

Now start with downloading the self-extracting provisioning script containing the minimal example SDWN core.
You will need to run the self-extracting provisioning script from the ESXi Shell, so enable ESXi Shell access.

The self-extracting provisioning script uses SSH to configure VMs. The ESXi firewall must therefore be configured to allow outbound SSH connections.

```bash
# esxcli network firewall ruleset set \
    --ruleset-id sshClient --enabled true
```

The self-extracting provisioning script also needs to temporarily store VMDKs and other large files during the installation process. The ESXi /tmp file system may have insufficient capacity to hold these files. We therefore recommend that you create a temporary directory under a mount point with at least 2 GB of spare capacity, and set the TMPDIR environment variable to point to this directory.

```bash
# mkdir /vmfs/volumes/datastore/tmp

# export TMPDIR=/vmfs/volumes/datastore/tmp
```

You will also need to specify vSwitch port groups for the external interfaces of the example core. It is possible to specify three different port groups; a management plane port group (where the management VM will receive an IP address by DHCP) a control plane port group (where ip1 is available for use) and a data plane port group (which is unused in the base configuration of the minimal example core). You should configure these port groups e.g. through the vSphere Client before running the provisioning script.

Finally, don't forget to backup your data and ESXi configuration before proceeding.

**Instructions**

Simply run the self-extracting provisioning script in the ESXi Shell.

```bash
# sh provision-anyfi-sdwn-core-r1f.sh minimal vmware_esxi \
    --datastore=datastore --with-mgmt-pg=lan \
    --with-ctrl-pg=wan --with-data-pg=wan
```

Note that the script can take up to 30 minutes to complete. Do not terminate the script during this time. Once the example core is provisioned start up the virtual machines and complete the installation as detailed below.
Automated Provisioning on VirtualBox

In this section we explain how to use the self-extracting provisioning script to provision a minimal example core on a Linux host system with Oracle VM VirtualBox hypervisor.

Preparations

First of all download the self-extracting provisioning script containing the minimal example SDWN core.

You will need a recent version of Oracle VM VirtualBox. This example core has been verified with version 4.3.10.

Instructions

Simply run the self-extracting provisioning script in the shell.

```
# sh provision-anyfi-sdwn-core-r1f.sh minimal virtualbox \
   --with-ctrl-if=eth0 --with-data-if=eth0
```

Note that the script can take up to 30 minutes to complete. Do not terminate the script during this time. Once the example core is provisioned start up the virtual machines and complete the installation as detailed below.

Manual Provisioning on VirtualBox

For Mac OS or Windows hosts we recommend manually installing the minimal example SDWN core using the Open Virtual Appliance (OVA) file.

Preparations

First of all download the Open Virtual Appliance (OVA) containing the minimal example SDWN core.

To manually install the Open Virtual Appliance (OVA) you will need a recent version of Oracle VM VirtualBox. This example core has been verified with version 4.3.10.

Instructions

We now go through the installation of the Open Virtual Appliance on Oracle VM VirtualBox step-by-step.
**Import the Open Virtual Appliance into Oracle VM VirtualBox**

Start VirtualBox and select the File > Import Appliance... menu option. Select the OVA file and press the Continue button. As you will note the OVA file contains multiple VMs, one for each network element in the Software-Defined Wireless Networking (SDWN) core. Import all the VMs into VirtualBox by pressing the Import button (without changing any settings).

**Select External Network Interfaces**

Select the mini-anyfi-ctrl VM and press Network. Select Adapter 2 and verify that it is configured to use a reasonable external network interface. We will later configure the VM to use ip1 for this interface.

**Start the Virtual Machines**
Select both VMs and press Start. Make sure they both boot and then minimize their console windows. We will not be using the console to interact with the VMs.

Completing the Installation

Once all VMs have booted you should be able to log in to the management VM through Secure Shell (SSH). Under VirtualBox the management VM is reachable at localhost port 8022. With VMware ESXi the management VM is attached to the port group you specified using the --with-mgmt-pg= option and will request an IP address through DHCP. In both cases you should log in as the user admin with password admin.

Log in to the management VM with SSH

```
host:~$ ssh -l admin [localhost -p 8022 | x.x.x.x]
```

Once logged in complete the installation of the core by issuing the following command.

Complete the installation of the example core

```
admin@mgmt:~$ ./install-core
```
Chapter 4: Configuration

In this chapter we describe how to configure your newly installed virtualized Software-Defined Wireless Networking (SDWN) core.

Configuring the Controller

We start by connecting to the Controller with Secure SHell (SSH).

<table>
<thead>
<tr>
<th>Connect to the Controller with SSH</th>
</tr>
</thead>
<tbody>
<tr>
<td>admin@mgmt:~$ ssh anyfi-ctrl</td>
</tr>
<tr>
<td>vyatta@anyfi-ctrl:~$</td>
</tr>
</tbody>
</table>

The Controller is based on the Vyatta Network OS and provides a Command Line Interface (CLI) for administrative tasks.

Basic Networking

The Controller acts as a rendezvous point for all other elements in the SDWN architecture and therefore must have a fixed (and usually public) IP address. We refer to this address as ip1. You will later need to configure it (or the corresponding domain name) into access points and residential gateways, so make note of it.

<table>
<thead>
<tr>
<th>Enter configuration mode</th>
<th>vyatta@anyfi-ctrl:~$ configure [edit]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configure basic networking</td>
<td>vyatta@anyfi-ctrl# set interfaces ethernet eth1 address ip1/xx [edit]</td>
</tr>
<tr>
<td></td>
<td>vyatta@anyfi-ctrl# set system name-server x.x.x.x [edit]</td>
</tr>
<tr>
<td></td>
<td>vyatta@anyfi-ctrl# set system gateway-address x.x.x.x [edit]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Commit, save and exit configuration mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>vyatta@anyfi-ctrl# commit</td>
</tr>
<tr>
<td>vyatta@anyfi-ctrl# save</td>
</tr>
<tr>
<td>vyatta@anyfi-ctrl# exit</td>
</tr>
</tbody>
</table>

The Controller should now have basic IP connectivity and name resolution. You can verify this e.g. with the ping command.

SDWN Controller

Now we are ready to configure the SDWN controller.
Enter configuration mode
vyatta@anyfi-ctrl:~$ configure
[edit]

Define client, radio and service groups
vyatta@anyfi-ctrl# edit service anyfi controller
[edit service anyfi controller]
vyatta@anyfi-ctrl# set client-group "all"
[edit service anyfi controller]
vyatta@anyfi-ctrl# set radio-group "all"
[edit service anyfi controller]
vyatta@anyfi-ctrl# set service-group "all"
[edit service anyfi controller]
vyatta@anyfi-ctrl# top
[edit]

Create an instance of the SIMPLE app
vyatta@anyfi-ctrl# edit service anyfi controller app simple "simple1"
[edit service anyfi controller app simple simple1]
vyatta@anyfi-ctrl# set clients "all"
[edit service anyfi controller app simple simple1]
vyatta@anyfi-ctrl# set radios "all"
[edit service anyfi controller app simple simple1]
vyatta@anyfi-ctrl# set services "all"
[edit service anyfi controller app simple simple1]
vyatta@anyfi-ctrl# top
[edit]

Review changes
vyatta@anyfi-ctrl# show service anyfi controller
+app {
+  simple simple1 {
+    clients all
+    radios all
+    services all
+  }
+}
+client-group all {
+}
+radio-group all {
+}
+service-group all {
+}
[edit]

Commit, save and exit configuration mode
vyatta@anyfi-ctrl# commit
vyatta@anyfi-ctrl# save
vyatta@anyfi-ctrl# exit

The Controller should now be configured and ready to use.

Disconnect from the Controller
vyatta@anyfi-ctrl:~$ exit
admin@mgmt:~$

This concludes the configuration of the minimal example SDWN core.
Chapter 5: Verification

In this chapter we describe how to put your newly installed and configured Software-Defined Wireless Networking (SDWN) core to good use.

Preparations

To follow the instructions below you will need two Anyfi.net enabled residential gateways. We will use Inteno VG50, a dual-WAN Ethernet and ADSL2+ residential gateway that is available from the manufacturer with Anyfi.net software pre-installed.

**NOTE** Anyfi.net software can easily be installed on any Wi-Fi router that runs OpenWrt. Please see [http://anyfi.net/openwrt/INSTALL](http://anyfi.net/openwrt/INSTALL) for step-by-step instructions.

Figure 8: The Inteno VG50 comes with Anyfi.net software pre-installed and the Controller IP address or fully qualified domain name can be configured through the WEBUI admin interface.

Log into the VG50’s WEBUI as the “admin” user and navigate to the Wi-Fi configuration page. Once there select the Anyfi.net tab under Device Configuration. Enter ip1 into the Controller field. Press the Save button. Since the Controller acts as a rendezvous point for all other network elements you do not need to configure any other IP addresses.

Repeat the process for the other VG50, and ensure that they both have IP connectivity on the WAN port. Then verify that they have both registered successfully with the Controller using the operational commands below.
Connect to the Controller

admin@mgmt:~$ ssh anyfi-ctrl
vyatta@anyfi-ctrl:~$

List connected radios

vyatta@anyfi-ctrl:~$ show anyfi controller radios
===================================================================
<table>
<thead>
<tr>
<th>hwaddr</th>
<th>ip</th>
<th>groups</th>
</tr>
</thead>
<tbody>
<tr>
<td>00:22:07:67:b2</td>
<td>118.13.81.8:52814</td>
<td>all</td>
</tr>
</tbody>
</table>

List connected services

vyatta@anyfi-ctrl:~$ show anyfi controller services
===================================================================
<table>
<thead>
<tr>
<th>uuid</th>
<th>ssid</th>
<th>auth</th>
<th>groups</th>
</tr>
</thead>
<tbody>
<tr>
<td>31fc0dc6-625f-1d8a-f86ab33b47c2</td>
<td>Inteno-D967B2</td>
<td>psk</td>
<td>other</td>
</tr>
<tr>
<td>5cbdclee-625f-1f8a-bc177e36f1b3</td>
<td>Inteno-D96914</td>
<td>psk</td>
<td>other</td>
</tr>
</tbody>
</table>

Disconnect from the Controller

vyatta@anyfi-ctrl:~$ exit
admin@mgmt:~$

Residential gateways should show up both as radios and as services in the lists above. You should also see the Gateway, but registered only as a service since it does not have a local Wi-Fi access point radio.

Instructions

We are now ready to try out the user experiences previously described, using our newly installed and configured Software-Defined Wireless Networking core.

SIMPLE Verification

Verify that a device previously connected to one of the VG50s will automatically connect to the same logical Wi-Fi network through the radio of the other, now as a guest user.

Note that the solution provides complete mobility with session continuity on Layer 3, even to and from the home Wi-Fi network. The user can thus start e.g. a Skype call in their own home, connected locally to the first VG50, and then walk out the door, being handed over to the other VG50 without disruption to the call.

Also note that while devices previously connected to one of the VG50s will continue to display the SSID of its home network even when connected through the other VG50, other devices will not detect this SSID. Each device sees only the networks that are relevant to them.
Chapter 6: Integration

In this chapter we describe how your verified SDWN core can be further integrated towards third party systems for CPE management and network monitoring. We exemplify by installing GenieACS and Splunk, both available for download free of charge, on the management VM. In a production environment these systems would however typically run independently of the SDWN core.

GenieACS for TR-069 CPE Management

GenieACS is an open source TR-069 Automatic Configuration Server (ACS) implementation flexible enough to support configuration of the Anyfi.net TR-069 vendor extension attributes without customization.

Before the management VM can serve as a TR-069 ACS it will need an external network interface on which it can be reached by CPEs under management. Add a network interface bridged to your external network and make note of a public IP address (ip5) to be used for the ACS.

Configure the management VM to use ip5 on its newly added external network interface.

Configure networking on management VM

admin@mgmt:~$ sudo vi /etc/network/interfaces

Review changes

admin@mgmt:~$ sudo cat /etc/network/interfaces

# The loopack network interface
auto lo
iface lo inet loopback

# Management
auto eth0
iface eth0 inet dhcp

# Management
auto eth1
iface eth1 inet static
  address 10.0.10.101
  network 10.0.10.0
  netmask 255.255.255.0

# WAN
auto eth2
iface eth2 inet static
  address ip6
  network x.x.x.x
  netmask x.x.x.x
  gateway x.x.x.x
  dns-nameservers x.x.x.x

Apply changes
admin@mgmt:~$ sudo ifup eth2

Now we are ready to install and start GenieACS. The management VM contains an installation script for the purpose.

Install GenieACS
admin@mgmt:~$ ./install-genieacs

Start GenieACS
admin@mgmt:~$ sudo service genieacs start

GenieACS comes with an administration WEBUI reachable at localhost port 3000. Use SSH port forwarding to access this interface from your host machine.

Log out of the management VM
admin@mgmt:~$ exit
host:~$

Log into the management VM with port forwarding
host:~$ ssh -l admin [localhost -p 8022 | x.x.x.x] -L 3001:localhost:3000
admin@mgmt:~$

The management web interface should now be accessible at http://localhost:3001.

Figure 9: The GenieACS welcome screen. Note the log in link in the top right corner.
Log in with the username "admin" and password "admin" and familiarize yourself with the user interface. If you have a CPE that implements the Anyfi.net TR-069 vendor extension connected to the ACS you can change its configuration by entering X_ANYFI_NET in the parameter search box and clicking Edit on the relevant parameter.

Figure 10: Screenshot of the GenieACS GUI when editing the Controller parameter.

The process of configuring Anyfi.net enabled devices can be automated using the preset functionality of GenieACS. Simply create a preset that unconditionally sets e.g. InternetGatewayDevice.X_ANYFI_NET_Config.Controller to ip1.
Splunk for Network Monitoring and Dashboards

Splunk is an operational intelligence and structured data analysis platform well suited for visualization and monitoring of radio link-level accounting information from the Controller.

The management VM contains an installation script for installing Splunk.

**Install Splunk**

```bash
admin@mgmt:$ ./install-splunk
```

**Start Splunk**

```bash
admin@mgmt:$ sudo service splunk start
```

Splunk’s WEBUI is reachable at localhost port 8000. Use SSH port forwarding to access this interface from your host machine.

**Log out of the management VM**

```bash
admin@mgmt:$ exit
```

**Log into the management VM with port forwarding**

```bash
host:$ ssh -l admin [localhost -p 8022 | x.x.x.x] -L 8001:localhost:8000

admin@mgmt:$
```


![Figure 11: The Anyfi Analytics app for Splunk provides basic dashboards for monitoring your SDWN network and can easily be extended with more specialized data analysis.](image)

---

MINIMAL SDWN CORE  GETTING STARTED GUIDE  R1F