Integrate the cnPilot enterprise Access Points with an external Guest Access Portal

Guest Access Portal Integration

cnPilot E400, E500, ePMP1000 Hotspot

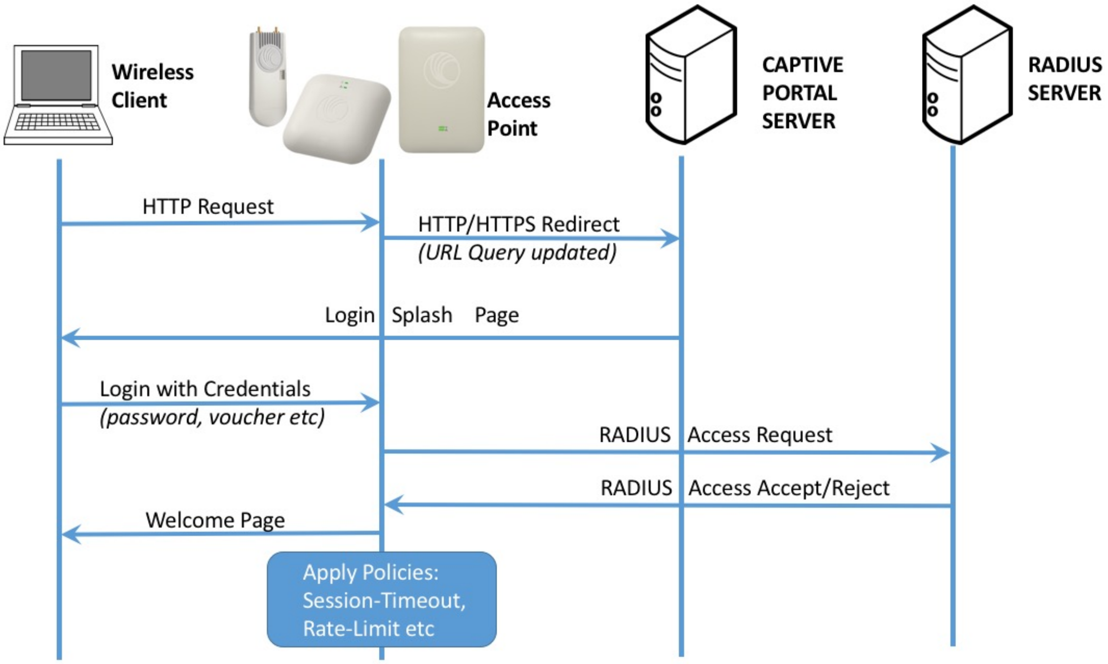
Table of Contents

Introduction

Cambium cnPilot E series access points such as the E400, E500, ePMP1000 Hotspot support guest access captive portals natively (built-in), but they can also be configured to provide guest-access services interoperating with an external server that provides portal services. This document describes the configuration and messaging details when the deployment uses an external 3rd party portal server.

Workflow

The general workflow when an external server is being used is as follows:



1. The client first associates with the AP and receives an IP address over DHCP. When the user tries to access some webpage (eg: http://www.cambiumnetworks.com) in the browser, the client would use DNS to resolve the IP address of that server, then attempt a TCP connection to the web servers IP address.
2. The AP will intercept this packet and redirect the user to the external captive portal server. As part of the redirection it will include some parameters into the URL query string which the external server can use to identify the client and process this request.
3. The external server will send back a splash page to the client where the user can then login.
4. The user would then login, the credentials would be posted to the Access Point

which would then look up the username/password on a RADIUS server, and based on the response from the server allow/deny access to the client.

Embedding Client Information in URL

Several attributes related to the users session are embedded into the URL the AP uses for the external redirection, the names of these attributes (query-strings) and their meanings are described in the following table:

ATTRIBUTE	DESCRIPTION
ga_ap_mac	The MAC address of the AP where the client is associated
ga_nas_id	The NAS-ID configured in the WLANs RADIUS configuration section. If none is configured, then the hostname of the AP
ga_srvr	Access Point Interface IP for the wireless client VLAN and if an external web server being used then the public interface IP address through which the external webserver can be reached out. This field can be used by the portal to connect to the AP for the client authentication/logout.
ga_cmac	MAC address of the wireless client
ga_orig_url	The original URL the client was trying to access, when it was redirected. Note: this is added only if the guest access config success action is to redirect to the original URL
ga_qv	This is a token used by the AP to complete the authentication process. It needs to be sent back unchanged to the AP, for it to successfully validate and process the clients login/logout request.
c_timeout	Added in the welcome URL as part of login success, indicates the session-time which is either configured on the WLAN or learnt through RADIUS attributes
ga_error_code	This is sent if the login request failed and has values like "timeout", "reject", "not-found". Our login page uses these codes to display respective error message to the end user.

Redirection URLs

1) Authentication module URL:

HTTP:

http://HOST:880/<WLAN_IDX>/login.html?ga_ap_mac=<AP_MAC>&ga_nas_id=<NAS_ID>&ga_srvr=<ACCESS_POINT_IP>&ga_cmac=<CLIENT_MAC>&ga_Qv=<ADDITIONAL_QUERY_STRINGS_USED_BY_ACCESS_POINT>

HTTPS:
https://HOST:444/<WLAN_IDX>/login.html?ga_ap_mac=<AP_MAC>&ga_nas_id=<NAS_ID>&ga_srvr=<ACCESS_POINT_IP>&ga_cmac=<CLIENT_MAC>&ga_Qv=<ADDITIONAL_QUERY_STRINGS_USED_BY_ACCESS_POINT>

2) Click-through module URL:

HTTP:
http://HOST:880/<WLAN_IDX>/terms.html?ga_ap_mac=<AP_MAC>&ga_nas_id=<NAS_ID>&ga_srvr=<ACCESS_POINT_IP>&ga_cmac=<CLIENT_MAC>&ga_Qv=<ADDITIONAL_QUERY_STRINGS_USED_BY_ACCESS_POINT>

HTTPS:
https://HOST:444/<WLAN_IDX>/terms.html?ga_ap_mac=<AP_MAC>&ga_nas_id=<NAS_ID>&ga_srvr=<ACCESS_POINT_IP>&ga_cmac=<CLIENT_MAC>&ga_Qv=<ADDITIONAL_QUERY_STRINGS_USED_BY_ACCESS_POINT>

URL parameters description:

a) The WLAN_IDX matches to the corresponding WLAN index in the access point configuration.
b) ga_ap_mac contains the access point MAC in "%02X-%02X-%02X-%02X-%02X-%02X" format.
c) ga_nas_id provides the configured nas-id or the default hostname of the access point.
d) ga_srvr provides the access point server IP address.
e) ga_cmac provides the wireless client mac address in the same format as the ga_ap_mac.
f) ga_Qv contains the encoded query strings which are used internally by the access point to perform the login action.

Note: In case the guest access configuration has been configured to redirect to user original URL after successful login then the above redirect URL will additionally have "&ga_orig_url=<ORIGINAL_CLIENT_URL>" in the redirect URL which is eventually used in the login response to give the original URL as the login success URL.

Example URL:

Login URL:
https://10.140.134.100:444/2/login.html?ga_ap_mac=00-04-56-C8-81-6B&ga_nas_id=kunal-ap.com&ga_srvr=10.140.134.100&ga_cmac=00-24-D7-C2-6B-

3C&ga_Qv=xDN%05%1F%3A%1C%18%00%181%04V%04%00%1B%0E%14Y%15%02%00%26%152%01O6%5EY-T_W%22F%5E%2AEU%7BYCQK%06%13%143%0C

Logout URL:
https://10.140.134.100:444/2/welcome.html?ga_ap_mac=00-04-56-C8-81-6B&ga_nas_id=kunal-ap.com&ga_srvr=10.140.134.100&ga_cmac=00-24-D7-C2-6B-3C&ga_Qv=xDN%05%1F%3A%1C%18%00%181%04V%04%00%1B%0E%14Y%15%02%00%26%152%01O6%5EY-T_W%22F%5E%2AEU%7BYCQK%06%13%143%0C&c_timeout=900

Web Pages Sample Source Code

Login Page

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8"><meta
name="viewport" content="width=device-width, maximum-scale=1.0, user-
scalable=yes" />
<link rel="stylesheet" type="text/css" href="../ga_style.css" />
</head>
<body>
<div class="login-card">
<title>Welcome to Cambium Powered Hotspot</title>
<h1>Welcome to Cambium Powered Hotspot</h1>
<p>Please enter username and password to get web access</p>
<form name="LoginPage" id="LoginPage" action="/cgi-bin/hotspot_login.cgi"
method="POST"><input type="text" placeholder="Username" name="ga_user"><input
type="password" placeholder="Password" name="ga_pass">
<div class="login-help" onClick="openClose('a1')" style="cursor:hand;
cursor:pointer"><b><ins>Terms and Agreement</ins>
</b><br><br>
</div>
<input type="submit" class="btn1" value="Login"><div
id='errmsg'></div></form><div id="a1" class="texter"><tr><td colspan="3"
<input type="hidden" name="next" value="1">
<textarea style="width:100%; font-size: 13px; background-color:
#F8F8F8;border:3x solid" rows="10" cols="100" name="disclaimer">Terms and
Conditions of Service Usage and Information Agreement
You agree not to use the Service to:
(a) Transmit any material that is unlawful, threatening, abusive, harassing,
tortuous, defamatory, obscene, libelous, and invasive of another's privacy,
racially, ethnically or otherwise objectionable;
```

```
(b) Harm, or attempt to harm, minors in any way;
(c) Transmit any material either by email, uploading, posting or otherwise
with intent to harass another, threatens or encourages bodily harm or
destruction of property.
(d) Impersonate any person or entity or falsely state or otherwise
misrepresent your affiliation with a person or entity; forge headers or
otherwise manipulate identifiers in order to disguise the origin of any
material transmitted through the Service;
(e) Transmit any material that you do not have a right to make available
under any law;
(f) Transmit any material that infringes any patent, trademark, trade secret,
copyright or other proprietary rights of any party;
(g) Transmit any unsolicited or unauthorized advertising, promotional
materials, "junk mail," "Spam," "chain letters," "pyramid schemes" or any
other form of solicitation or collect personal information without consent
(h) "Hack" or otherwise attempt unauthorized access or spoofing of any other
site or service.
(i) Use the Service for excessively high volume data transfers, hosting a web
server, reselling the Service, denial of service attacks, or interference
with any sites or services.
</textarea>
</div>
</div>
<script>function getParameterByName(name) {
    name = name.replace(/[\\[]/, "\\[").replace(/[\]]/, "\\]");
    var regex = new RegExp("[\?&]" + name + "=([^&#]*)"),
    results = regex.exec(location.search);
    return results === null ? "" :
decodeURIComponent(results[1].replace(/\+/g, " "));
}
error_code = getParameterByName("ga_error_code");
var elem = document.getElementById('errmsg');
if (error_code == "timeout") {
   elem.innerHTML = '<font color=red>Authentication server failed to respond,
retry again in few minutes</font>';
} else if (error_code == "reject") {
   elem.innerHTML = '<font color=red>The username or password you entered is
incorrect</font>';
} else if (error_code == "not-found") {
   elem.innerHTML = '<font color=red>The authentication server not available,
contact network admin</font>';
} else {
   elem.innerHTML = '';
}
<!--
if (document.getElementById) {
    document.writeln('<style type="text/css"><!--')
    document.writeln('.texter {display:none} @media print {.texter
{display:block;}}')
    document.writeln('//--></style>')
}
function openClose(theID) {
   if (document.getElementById(theID).style.display == "block") {
document.getElementById(theID).style.display = "none" }
   else { document.getElementById(theID).style.display = "block" }
}
// -->
</script>
</body>
</html>
```

Once the redirection is done the client will post the login request, which will go the external web portal server, which eventually should process the request and frame a new request in the given format to be posted to the access point. The post to access point is done to below URL with a very similar format as the redirection URL

Example HTTP Post Request:

```
POST /cgi-bin/hotspot_login.cgi HTTP/1.1
Host: 192.168.2.7:880
Content-Type: application/x-www-form-urlencoded
Origin: http://192.168.2.7:880
Content-Length: 28
Connection: keep-alive
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_10_4)
AppleWebKit/600.7.12 (KHTML, like Gecko) Version/8.0.7 Safari/600.7.12
Referer: http://192.168.2.7:880/3/login.html?ga_ap_mac=00-04-56-C8-FE-
72&ga_nas_id=kunal-ap.com&ga_srvr=192.168.2.7&ga_cmac=34-36-3B-D3-4B-
2C&ga_Qv=%7BDN%05%1F%3A%02%18%1A0%0DA%15%12%1F%16%09O%17%0B%1D%177%3E%0ED%2AG
%406%5BOA%21L%29ZC%5ERd%40%40%5C%15%04%1F%26%0F
Accept-Language: en-us
Accept-Encoding: gzip, deflate
```

HTTP POST content:

```
ga_user=user_name&ga_pass=user_password
```

The HTTPS post request is sent to port 444.

In case the login attempt failes then the HTTP response contains the same login URL prefixed by the error_code query string in the URL which is used by the login.html to display the corresponding error code. Hence the external webportal server should be able to handle and parse the new URL which now contains the error_code attached to it

Example URL with error code:

http://HOST:880/<WLAN_IDX>/terms.html?ga_ap_mac=<AP_MAC>&ga_nas_id=<NAS_ID>&ga_srvr=<ACCESS_POINT_IP>&ga_cmac=<CLIENT_MAC>&ga_Qv=<ADDITIONAL_QUERY_STRINGS_USED_BY_ACCESS_POINT>&ga_error_code=<error_code>

The error_code values can be "timeout", "reject" and "not-found".

If the login is successful then the HTTP response contains the welcome URL with few additional query strings attached to it like c_timeout.

Example welcome URL:

```
"http://HOST:880/<WLAN-IDX>/welcome.html?ga_ap_mac=<AP_MAC>&ga_nas_id=<NAS_ID>&ga_srvr=<ACCESS_POINT_IP>&ga_cmac=<CLIENT_MAC>&ga_Qv=<ADDITIONAL_QUERY_STRINGS>&c_timeout=<SESSION_TIME>"
```

```
"https://HOST:444/<WLAN-IDX>/welcome.html?ga_ap_mac=<AP_MAC>&ga_nas_id=<NAS_ID>&ga_srvr=<ACCESS_POINT_IP>&ga_cmac=<CLIENT_MAC>&ga_Qv=<ADDITIONAL_QUERY_STRINGS>&c_timeout=<SESSION_TIME>"
```

HTML response for the login request with the above welcome URL/LOGIN URL with error code added to it in case the login attempt failed.

```
<html>
<head>
<title>TITLE</title>
<meta http-equiv=\"refresh\" content=\"0; URL='WELCOME_URL'
<body>
</body>
</html>
```

If original URL used then the reply will be temporary redirect response to the original URL:

```
HTTP/1.1 302 Temporary Redirect
Content-Type: text/html
Connection: close
Location: <client_orig_url>
```

Welcome page

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8"><meta
name="viewport" content="width=device-width, maximum-scale=1.0, user-
```

```
scalable=yes" />
<link rel="stylesheet" type="text/css" href="../ga_style.css" />
</head>
<body>
<div class="login-card"><title>Welcome to Cambium Powered
Hotspot</title><h1>Welcome to Cambium Powered Hotspot</h1><p>Congratulations
your login is successful</p>
<form action="/cgi-bin/hotspot_logout.cgi" method="POST"><input type="submit"
class="btn1" value="Logout">
</form>
<span id="countdown" class="timer"></span></center>
</div>
<script language="JavaScript">
// Gets the timeout from the URL
function getTimeout() {
var rez = window.location.search.match(/c_timeout=(\d+)/)
if (rez&&rez.length == 2) {
return parseInt(rez[1],10)
}
return -1;  // No timeout found
}
var seconds = getTimeout();
function secondPassed() {
var minutes = Math.round((seconds - 30)/60);
var hours = Math.round((minutes - 30)/60);
var remainingSeconds = seconds % 60;
if (remainingSeconds < 10) {
remainingSeconds = "0" + remainingSeconds;
}
var remainingMinutes = minutes % 60;
if (remainingMinutes < 10) {
remainingMinutes = "0" + remainingMinutes;
}
var remainingHours = hours % 60;
if (remainingHours < 10) {
remainingHours = "0" + remainingHours;
}
document.getElementById('countdown').innerHTML = "Session time remaining: " +
remainingHours + ":" + remainingMinutes +  ":" + remainingSeconds;
if (seconds == 0) {
clearInterval(countdownTimer);
document.getElementById('countdown').innerHTML = "<font color=red>Session
Expired, relogin to get access</font>";
} else {
seconds--;
}
}
if (seconds != -1) {
var countdownTimer = setInterval('secondPassed()', 1000);
}
```

```
</script>
</body>
</html>
```

Click through terms

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8"><meta
name="viewport" content="width=device-width, maximum-scale=1.0, user-
scalable=yes" /><link rel="stylesheet" type="text/css" href="../ga_style.css"
/>
</head>
<body>
<div class="login-card"><title>Welcome to Cambium Powered Hotspot</title>
<h1>Welcome to Cambium Powered Hotspot</h1>
<div class="login-help" onClick="openClose('a1')" style="cursor:hand;
cursor:pointer"><center><b><ins>Terms and
Agreement</ins></b></center><br></div><div id="a1" class="texter"><tr><td
colspan="3" <input type="hidden" name="next" value="1"><textarea
style="width:100%; font-size: 13px; background-color: #F8F8F8;border:3x
solid" rows="10" cols="100" name="disclaimer">Terms and Conditions of Service
Usage and Information Agreement
You agree not to use the Service to:
(a) Transmit any material that is unlawful, threatening, abusive, harassing,
tortuous, defamatory, obscene, libelous, and invasive of another's privacy,
racially, ethnically or otherwise objectionable;
(b) Harm, or attempt to harm, minors in any way;
(c) Transmit any material either by email, uploading, posting or otherwise
with intent to harass another, threatens or encourages bodily harm or
destruction of property.
(d) Impersonate any person or entity or falsely state or otherwise
misrepresent your affiliation with a person or entity; forge headers or
otherwise manipulate identifiers in order to disguise the origin of any
material transmitted through the Service;
(e) Transmit any material that you do not have a right to make available
under any law;
(f) Transmit any material that infringes any patent, trademark, trade secret,
copyright or other proprietary rights of any party;
(g) Transmit any unsolicited or unauthorized advertising, promotional
materials, "junk mail," "Spam," "chain letters," "pyramid schemes" or any
other form of solicitation or collect personal information without consent
(h) "Hack" or otherwise attempt unauthorized access or spoofing of any other
site or service.
(i) Use the Service for excessively high volume data transfers, hosting a web
server, reselling the Service, denial of service attacks, or interference
with any sites or services.
```

```
</textarea></div><form name="LoginPage" id="LoginPage" action="/cgi-
bin/hotspot_login.cgi" method="POST"><input type="submit" class="btn1"
value="I Agree">
</form>
</div>
<script language="JavaScript" type="text/javascript">
<!--
if (document.getElementById) {
document.writeln('<style type="text/css"><!--')
document.writeln('.texter {display:none} @media print {.texter
{display:block;}}')
document.writeln('//--></style>')
}
function openClose(theID) {
if (document.getElementById(theID).style.display == "block") {
document.getElementById(theID).style.display = "none" }
else { document.getElementById(theID).style.display = "block" }
}
// -->
</script>
</body>
</html>
```

RADIUS Attributes Supported

Access Requests

The following RADIUS attributes are supported in RADIUS Access-Request messages:

RADIUS Attribute	What the AP fills in
FRAMED-IP-ADDRESS	Wireless client IP address
NAS-IP-ADDRESS	IP Address of the AP
CALLED-STATION-ID	MAC address of the AP and the SSID where it is mapped, formatted as "MAC:SSID"
NAS-IDENTIFIER	Configured NAS-Identified of the WLAN. If not set, then the hostname is used.
NAS-PORT-ID	The SSID of the WLAN
NAS-PORT-TYPE	802.11
CALLING-STATION-ID	MAC address of the client

Access Accept

The following RADIUS attributes are understood by the AP in Access-Accept Messages:

RADIUS Attribute	How the AP uses this value
SESSION-TIMEOUT	Number of seconds this session is valid
IDLE-TIMEOUT	Number of seconds of no traffic from/to the client before it is disconnected for inactivity.
ACCT_INTERIM_INTVL	Interim Accounting Interval
CLASS	Cached by the AP and used as-is in the Accounting Request packets
REPLY-MESSAGE	Shown to the client when access is rejected
WIFI_ALLIANCE_MAX_UP	Upstream Rate-limit of client traffic
WIFI_ALLIANCE_MAX_DOWN	Downstream rate-limit of client traffic