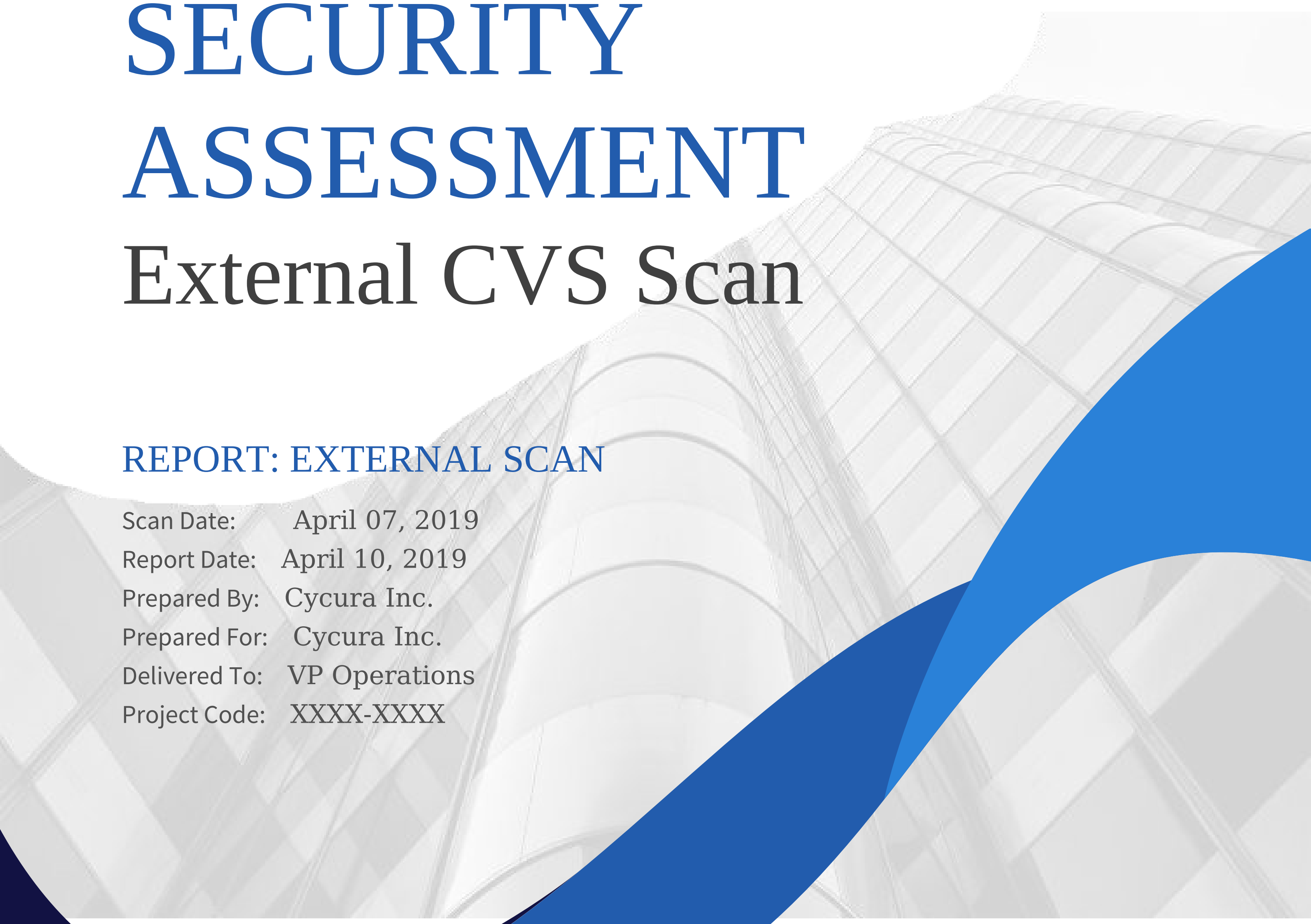
SECURITY ASSESSMENT

External CVS Scan



REPORT: EXTERNAL SCAN

Scan Date:	April 07, 2019
Report Date:	April 10, 2019
Prepared By:	Cycura Inc.
Prepared For:	Cycura Inc.
Delivered To:	VP Operations
Project Code:	XXXX-XXXX

CYCURA™

TABLE OF CONTENTS

TABLE OF CONTENTS....4
1 | PLACEHOLDER....6
1.1 Placeholder....6
1.1.1 Placeholder....6
1.2 Placeholder....6
1.2.1 Placeholder....6
1.3 Placeholder....7
1.3.1 Placeholder....7
1.3.2 Placeholder....7
1.4 Placeholder....7
1.4.1 Placeholder....8
1.4.2 Placeholder....8
1.5 Placeholder....8
1.5.1 Placeholder....8
2 | PLACEHOLDER....9
2.1 Placeholder....9
2.1.1 Placeholder....9
2.1.1.1 Placeholder....9
2.2 Placeholder....9
3 | PLACEHOLDER....10
3.1 Placeholder....10
3.1.1 Placeholder....10
3.1.1.1 Placeholder....10
3.2 Placeholder....10
4 | PLACEHOLDER....11
4.1 Placeholder....11
4.1.1 Placeholder....11
4.1.1.1 Placeholder....11
4.2 Placeholder....11
5 | PLACEHOLDER....12
5.1 Placeholder....12

5.1.1 Placeholder..12
5.1.1.1 Placeholder..12
5.2 Placeholder..12
5.3 Placeholder..13

1 | EXECUTIVE SUMMARY

1.1 Introduction

Cycura was engaged by Cycura Inc. (CYCURA) to perform an External CVS Scan against their external infrastructure. These assessments were performed during the month of April 2019.

1.2 Security Scorecard

Following analysis of all data obtained during this assessment, it is Cycura's opinion that the overall security rating of CYCURA's computing environment is **GOOD**.

1.2.1 Security Ratings

Every vulnerability on a system can expose information. Hostile parties use these security holes to steal, modify, or delete data. Past examples have included thefts of customer lists, and deletion of billing data. If account information is exposed, hackers can use it to gain control over a host either by exposing more information or using the host as a staging ground to launch an attack on yet another system. Being used as an accomplice in this way could expose CYCURA to legal action from the final victim in the chain.

Critical vulnerabilities are defined as those that can, independent of any other vulnerability, lead to a host being compromised. A single critical vulnerability is unacceptable. These vulnerabilities should be checked in more detail and corrected as soon as possible.

High Risk vulnerabilities usually require additional information or vulnerabilities (medium, or several low vulnerabilities) before they can be used to compromise the host. Cycura generally regards denial of service attacks as high risk vulnerabilities. These vulnerabilities should be checked in more detail and corrected as soon as possible.

Medium Risk and Low Risk vulnerabilities can make a host more susceptible to compromise. Providing unnecessary information about the host or its operating environment would be considered a low risk. It is recommended that these vulnerabilities be checked in more detail and corrected if appropriate.

1.2.2 Common Vulnerability Exposures and Scoring System

Cycura incorporates two systems for tracking and scoring common vulnerabilities. Common Vulnerabilities and Exposures (CVE) and Common Vulnerability Scoring System (CVSS) are the standards included in this report.

CVE is a dictionary of publicly known information security vulnerabilities and exposures. This unique identifier provides patch information, vendor disclosures and any known public exploits.

CVSS is a vulnerability scoring system designed to provide an open and standardized method for rating IT vulnerabilities. CVSS helps organizations prioritize and coordinate a joint response to security vulnerabilities by communicating the base, temporal and environmental properties of a vulnerability.

1.3 Objectives and Scope

1.3.1 Objectives

The main objectives of this project are to:

- Review CYCURA's external infrastructure; and
- Provide a summary of concerns and approaches that might be used to remedy identified risk issues.

1.3.2 Scope

External vulnerability assessment testing scope:

- 104.236.88.120

1.4 Key Findings

The identified vulnerabilities are categorized according to their potential to allow harm or unauthorized access to the target system. Risk factor vulnerabilities are categorized as follows:

Severity	Vulnerability
MEDIUM	Apache Server Multiple Vulnerabilities
MEDIUM	SSL Certificate Cannot Be Trusted
MEDIUM	SSH Weak Algorithms Supported
MEDIUM	SSL Self-Signed Certificate
MEDIUM	SSL Medium Strength Cipher Suites Supported (SWEET32)
LOW	SSH Weak MAC Algorithms Enabled
LOW	SSH Server CBC Mode Ciphers Enabled
LOW	SSL RC4 Cipher Suites Supported (Bar Mitzvah)

1.5 Key Recommendations

[ENTER KEY RECOMMENDATIONS HERE]

1.6 Conclusion

Cycura would like to thank all CYCURA staff members who assisted us in the process of performing this security analysis. While there are a number of medium-severity items, which need to be resolved to better secure the operationally critical infrastructure, the security meets with accepted standards for "Industry Best Practice". The overall posture was found to be strong with only a handful of medium and low risk shortcomings identified.

Looking forward, Cycura recommends that CYCURA use this document as a method to gauge future performance and as a basis for further risk and vulnerability analysis.

NOTE: This determination of relative security and absence of other vulnerabilities represents a point-in-time statement current at the time that testing was conducted; it may not represent future conditions.

2 | Findings

Medium Findings

Apache Server Multiple Vulnerabilities

Vulnerability Rating: Medium (CVSS Score: 5.0)

The identified hosts are running outdated version of Apache HTTP Server, which is reportedly vulnerable to multiple issues, such as denial of service, code execution, cross-site scripting, information disclosure and others.

Recommendations:

- Update to latest stable version.

CVEs:

- CVE-2001-0731

Affected Hosts:

- 104.236.88.120

Evidence for 104.236.88.120:

```
Cycura was able to exploit the issue using the following request :

http://104.236.88.120:443/?M=A



This produced the following truncated output (limited to 10 lines) :
------------------------------ snip ------------------------------
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<html>
<head>
<title>Index of /</title>
```

```
</head>
<body>
<h1>Index of /</h1>
<table>
<tr><th valign="top"><script data-pagespeed-no-defer
type="text/javascript">//<![CDATA[
(function(){var g=this;function h(b,d){var a=b.split("."),c=g;a[0]in c||!
c.execScript||c.execScript("var "+a[0]);for(var
e;a.length&&(e=a.shift());)a.length||void 0===d?c[e]?
c=c[e]:c=c[e]={}:c[e]=d};function l(b){var d=b.length;if(0<d){for(var
a=Array(d),c=0;c<d;c++)a[c]=b[c];return a}return[]};function m(b){var
d=window;if(d.addEventListener)d.addEventListener("load",b,!1);else
if(d.attachEvent)d.attachEvent("onload",b);else{var
a=d.onload;d.onload=function(){b.call(this);a&&a.call(this)}}};var
n;function p(b,d,a,c,e)
{this.h=b;this.j=d;this.l=a;this.f=e;this.g={height:window.innerHeight||
document.documentElement.clientHeight||
document.body.clientHeight,width:window.innerWidth||
document.documentElement.clientWidth||
document.body.clientWidth};this.i=c;this.b={};this.a=[];this.c={}}functio
n q(b,d){var a,c,e=d.getAttribute("data-pagespeed-url-hash");if(a=e&&!(e
in b.c))if(0>=d.offsetWidth&&0>=d.offsetHeight)a=!
1;else{c=d.getBoundingClientRect();var f=document.body;a=c.top+
("pageYOffset"in window?window.pageYOffset:(document.documentElement||
f.parentNode||f).scrollTop);c=c.left+("pageXOffset"in window?
window.pageXOffset:(document.documentElement||f.parentNode||
f).scrollLeft);f=a.toString()+","+c;b.b.hasOwnProperty(f)?a=!1:(b.b[f]=!
0,a=a<=b.g.height&&c<=b.g.width)}a&&(b.a.push(e),b.c[e]=!
0)}p.prototype.checkImageForCriticality=function(b)
{b.getBoundingClientRect&&q(this,b)};h("pagespeed.CriticalImages.checkIma
geForCriticality",function(b)
{n.checkImageForCriticality(b)});h("pagespeed.CriticalImages.checkCritica
lImages",function(){r(n)});function r(b){b.b={};for(var
d=["IMG","INPUT"],a=[],c=0;c<d.length;+
+c)a=a.concat(l(document.getElementsByTagName(d[c])));if(0!
=a.length&&a[0].getBoundingClientRect){for(c=0;d=a[c];+
+c)q(b,d);a="oh="+b.l;b.f&&(a+="&n="+b.f);if(d=0!
=b.a.length)for(a+="&ci="+encodeURIComponent(b.a[0]),c=1;c<b.a.length;+
+c){var
e=","+encodeURIComponent(b.a[c]);131072>=a.length+e.length&&(a+=e)}b.i&&(
e="&rd="+encodeURIComponent(JSON.stringify(t())),131072>=a.length+e.lengt
h&&(a+=e),d=!0);u=a;if(d){c=b.h;b=b.j;var
f;if(window.XMLHttpRequest)f=new XMLHttpRequest;else
if(window.ActiveXObject)try{f=new
ActiveXObject("Msxml2.XMLHTTP")}catch(k){try{f=new
ActiveXObject("Microsoft.XMLHTTP")}catch(v){}}f&&(f.open("POST",c+(-
1==c.indexOf("?")?"?":"&")
+"url="+encodeURIComponent(b)),f.setRequestHeader("Content-
Type","application/x-www-form-urlencoded"),f.send(a))}}}function t(){var
b={},d=document.getElementsByTagName("IMG");if(0==d.length)return{};var
a=d[0];if(!("naturalWidth"in a&&"naturalHeight"in a))return{};for(var
c=0;a=d[c];++c){var e=a.getAttribute("data-pagespeed-url-hash");e&&(!(e
in b)&&0<a.width&&0<a.height&&0<a.naturalWidth&&0<a.naturalHeight||e in
b&&a.width>=b[e].o&&a.height>=b[e].m)&&(b[e]={rw:a.width,rh:a.height,ow:a
.naturalWidth,oh:a.naturalHeight})}return b}var
u="";h("pagespeed.CriticalImages.getBeaconData",function(){return
u});h("pagespeed.CriticalImages.Run",function(b,d,a,c,e,f){var k=new
p(b,d,a,e,f);n=k;c&&m(function(){window.setTimeout(function()
{r(k)},0)})});})();pagespeed.CriticalImages.Run('/
mod_pagespeed_beacon','http://104.236.88.120:443/?
M=A','HxFJjT4Vi3',true,false,'gPcm8tVgTj8');
[...]

------------------------------ snip ------------------------------
```

SSL Certificate Cannot Be Trusted

Vulnerability Rating: Medium (CVSS Score: 6.4)

The server's X.509 certificate cannot be trusted. This situation can occur in three different ways, in which the chain of trust can be broken, as stated below :

- First, the top of the certificate chain sent by the server might not be descended from a known public certificate authority. This can occur either when the top of the chain is an unrecognized, self-signed certificate, or when intermediate certificates are missing that would connect the top of the certificate chain to a known public certificate authority.

- Second, the certificate chain may contain a certificate that is not valid at the time of the scan. This can occur either when the scan occurs before one of the certificate's 'notBefore' dates, or after one of the certificate's 'notAfter' dates.

- Third, the certificate chain may contain a signature that either didn't match the certificate's information or could not be verified. Bad signatures can be fixed by getting the certificate with the bad signature to be re-signed by its issuer. Signatures that could not be verified are the result of the certificate's issuer using a signing algorithm that Nessus either does not support or does not recognize.

If the remote host is a public host in production, any break in the chain makes it more difficult for users to verify the authenticity and identity of the web server. This could make it easier to carry out man-in-the-middle attacks against the remote host.

Recommendations:

- Purchase or generate a proper certificate for this service.

Resources:

- https://www.itu.int/rec/T-REC-X.509/en
- https://en.wikipedia.org/wiki/X.509

Affected Hosts:

- 104.236.88.120

Evidence for 104.236.88.120:

```

The following certificate was at the top of the certificate
chain sent by the remote host, but it is signed by an unknown
certificate authority :

|-Subject :
C=CA/ST=Ontario/L=Toronto/O=WildRound/CN=hosting.wildround.com/E=spam@wil
dround.com
|-Issuer  :
C=CA/ST=Ontario/L=Toronto/O=WildRound/CN=hosting.wildround.com/E=spam@wil
dround.com

```

SSH Weak Algorithms Supported

Vulnerability Rating: Medium (CVSS Score: 4.3)

Nessus has detected that the remote SSH server is configured to use the Arcfour stream cipher or no cipher at all. RFC 4253 advises against using Arcfour due to an issue with weak keys.

Recommendations:

- Contact the vendor or consult product documentation to remove the weak
- ciphers.

Resources:

- https://tools.ietf.org/html/rfc4253#section-6.3

Affected Hosts:

- 104.236.88.120

Evidence for 104.236.88.120:

```
The following weak server-to-client encryption algorithms are supported :



  arcfour
  arcfour128
  arcfour256

The following weak client-to-server encryption algorithms are supported :



  arcfour
  arcfour128
  arcfour256
```

SSL Self-Signed Certificate

Vulnerability Rating: Medium (CVSS Score: 6.4)

The X.509 certificate chain for this service is not signed by a recognized certificate authority. If the remote host is a public host in production, this nullifies the use of SSL as anyone could establish a man-in-the-middle attack against the remote host.

Note that Cycura does not check for certificate chains that end in a certificate that is not self-signed, but is signed by an unrecognized certificate authority.

Recommendations:

- Purchase or generate a proper certificate for this service.

Affected Hosts:

- 104.236.88.120

Evidence for 104.236.88.120:

```
The following certificate was found at the top of the certificate
chain sent by the remote host, but is self-signed and was not
found in the list of known certificate authorities :

|-Subject :
C=CA/ST=Ontario/L=Toronto/O=WildRound/CN=hosting.wildround.com/E=spam@wil
dround.com
```

SSL Medium Strength Cipher Suites Supported (SWEET32)

Vulnerability Rating: Medium (CVSS Score: 5.0)

The remote host supports the use of SSL ciphers that offer medium strength encryption. Nessus regards medium strength as any encryption that uses key lengths at least 64 bits and less than 112 bits, or else that uses the 3DES encryption suite.

Note that it is considerably easier to circumvent medium strength encryption if the attacker is on the same physical network.

Recommendations:

- Reconfigure the affected application if possible to avoid use of
- medium strength ciphers.

Resources:

- https://www.openssl.org/blog/blog/2016/08/24/sweet32/
- https://sweet32.info

CVEs:

- CVE-2016-2183

Affected Hosts:

- 104.236.88.120

Evidence for 104.236.88.120:

```
  Medium Strength Ciphers (> 64-bit and < 112-bit key, or 3DES)

    EDH-RSA-DES-CBC3-SHA         Kx=DH          Au=RSA      Enc=3DES-
CBC(168)        Mac=SHA1
    ECDHE-RSA-DES-CBC3-SHA       Kx=ECDH        Au=RSA      Enc=3DES-
CBC(168)        Mac=SHA1
    DES-CBC3-SHA                 Kx=RSA         Au=RSA      Enc=3DES-
```

```
CBC(168)        Mac=SHA1

The fields above are :

  {OpenSSL ciphername}
  Kx={key exchange}
  Au={authentication}
  Enc={symmetric encryption method}
  Mac={message authentication code}
  {export flag}
```

Low Findings

SSH Weak MAC Algorithms Enabled

Vulnerability Rating: Low (CVSS Score: 2.6)

The remote SSH server is configured to allow either MD5 or 96-bit MAC algorithms, both of which are considered weak.

Note that Cycura only checks for the options of the SSH server,
and it does not check for vulnerable software versions.

Recommendations:

- Contact the vendor or consult product documentation to disable MD5 and
- 96-bit MAC algorithms.

Affected Hosts:

- 104.236.88.120

Evidence for 104.236.88.120:

```
The following client-to-server Message Authentication Code (MAC)
algorithms
are supported :

  hmac-md5
  hmac-md5-96
  hmac-md5-96-etm@openssh.com
  hmac-md5-etm@openssh.com
  hmac-sha1-96
  hmac-sha1-96-etm@openssh.com

The following server-to-client Message Authentication Code (MAC)
algorithms
are supported :
```

```
hmac-md5
hmac-md5-96
hmac-md5-96-etm@openssh.com
hmac-md5-etm@openssh.com
hmac-sha1-96
hmac-sha1-96-etm@openssh.com
```

SSH Server CBC Mode Ciphers Enabled

Vulnerability Rating: Low (CVSS Score: 2.6)

The SSH server is configured to support Cipher Block Chaining (CBC) encryption. This may allow an attacker to recover the plaintext message from the ciphertext.

Note that Cycura only checks for the options of the SSH server and does not check for vulnerable software versions.

Recommendations:

- Contact the vendor or consult product documentation to disable CBC mode
- cipher encryption, and enable CTR or GCM cipher mode encryption.

CVEs:

- CVE-2008-5161

Affected Hosts:

- 104.236.88.120

Evidence for 104.236.88.120:

```

The following client-to-server Cipher Block Chaining (CBC) algorithms
are supported :

  3des-cbc
  aes128-cbc
  aes192-cbc
  aes256-cbc
  blowfish-cbc
  cast128-cbc
  rijndael-cbc@lysator.liu.se

The following server-to-client Cipher Block Chaining (CBC) algorithms
are supported :
```

```
3des-cbc
aes128-cbc
aes192-cbc
aes256-cbc
blowfish-cbc
cast128-cbc
rijndael-cbc@lysator.liu.se
```

SSL RC4 Cipher Suites Supported (Bar Mitzvah)

Vulnerability Rating: Low (CVSS Score: 2.6)

The remote host supports the use of RC4 in one or more cipher suites. The RC4 cipher is flawed in its generation of a pseudo-random stream of bytes so that a wide variety of small biases are introduced into the stream, decreasing its randomness.

If plaintext is repeatedly encrypted (e.g., HTTP cookies), and an attacker is able to obtain many (i.e., tens of millions) ciphertexts, the attacker may be able to derive the plaintext.

Recommendations:

- Reconfigure the affected application, if possible, to avoid use of RC4
- ciphers. Consider using TLS 1.2 with AES-GCM suites subject to browser
- and web server support.

Resources:

- http://cr.yp.to/talks/2013.03.12/slides.pdf
- http://www.isg.rhul.ac.uk/tls/
- https://www.imperva.com/docs/HII_Attacking_SSL_when_using_RC4.pdf

CVEs:

- CVE-2015-2808

Affected Hosts:

- 104.236.88.120

Evidence for 104.236.88.120:

```
List of RC4 cipher suites supported by the remote server :

  High Strength Ciphers (>= 112-bit key)
```

```
    ECDHE-RSA-RC4-SHA            Kx=ECDH       Au=RSA      Enc=RC4(128)
Mac=SHA1
    RC4-SHA                      Kx=RSA        Au=RSA      Enc=RC4(128)
Mac=SHA1

The fields above are :

  {OpenSSL ciphername}
  Kx={key exchange}
  Au={authentication}
  Enc={symmetric encryption method}
  Mac={message authentication code}
  {export flag}
```

3 | Methodology

3.1 Placeholder

Cycura employs a tested and proven methodology when performing vulnerability assessments. Our methodology incorporates the Open Source Testing Methodology Manual (www.osstmm.org) and OWASP Top 10 (www.owasp.org). The goal of this methodology is to identify risks to the assets under assessment and ultimately the underlying business.

3.1.1 Approach

A Black Box approach was used in this assessment, with no prior knowledge of the target system, save for a valid IP range. Domain admin credentials were not supplied to the testing team for the external assessment.

3.1.1 Assessment Phases

The methodology may vary depending on the target scope and restrictions set by the client. However, the primary phases are:

Discovery phase, in which information is gathered on the target organization through websites, mail servers, public records and databases (Address and Name Registrars, DNS, Whois, EDGAR, etc.)

Enumeration phase, in which the penetration team actively tries to obtain user names, network shares and application version information of running services.

Vulnerability Scanning phase, in which the test team maps the profile of the environment to identify publicly known vulnerabilities and previously undiscovered issues. Password bruteforce tests are also performed using special dictionaries maintained by Cycura. This phase is executed using automated commercial and open-source testing tools, and manual testing by experienced penetration testers.

Exploitation phase, in which the test team will try to gain privileged access to a target system by directly exploiting or chaining more than one of the vulnerabilities identified in the previous step.