**Stage 2**

**Report**

**Target Website:** https://ritindia.edu/

**Target IP Address:** 68.178.151.68

#### Vulnerability Name: PHP Unsupported Version Detection

#### Severity: Critical

#### Plug-in: #58987

**Port:** 443 / tcp / www

**Description:** According to its version, the installation of PHP on the remote host is no longer supported.  
  
Lack of support implies that no new security patches for the product will be released by the vendor. As a result, it is likely to contain security vulnerabilities.

**Solution:** Upgrade to a version of PHP that is currently supported.

**Business Impact:** Anyone can connect to the NSClient and retrieve sensitive information, such as process and service states, memory usage, etc.

#### Vulnerability Name: SSL Anonymous Cipher Suites Supported

#### Severity: Medium

#### Plug-in: #31705

**Port:** 21 / tcp / ftp

**Description:** The remote host supports the use of anonymous SSL ciphers. While this enables an administrator to set up a service that encrypts traffic without having to generate and configure SSL certificates, it offers no way to verify the remote host's identity and renders the service vulnerable to a man-in-the-middle attack.  
  
Note: This is considerably easier to exploit if the attacker is on the same physical network..

**Solution:** Reconfigure the affected application if possible to avoid use of weak ciphers.

**Business Impact:** Cipher suites, like any other cryptographic technology, aren’t foolproof. As technology advances, vulnerabilities may emerge, making some suites susceptible to attacks. Human errors, outdated protocols, or poor implementations can also compromise security.

#### Vulnerability Name: SSL Medium Strength Cipher Suites Supported (SWEET32)

#### Severity: High

#### Plug-in: #42873

**Port:** 21 / tcp / ftp

**Description:** 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.

**Solution:** Reconfigure the affected application if possible to avoid use of medium strength ciphers.

#### Vulnerability Name: SSL Certificate Cannot Be Trusted

#### Severity: Medium

#### Plug-in: #51192

**Port:** 443 / tcp / www

**Description:** 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.

**Solution:** Purchase or generate a proper SSL certificate for this service.

Business Impact:

#### SSL Certificate Cannot Be Trusted

#### Vulnerability Name: SSL RC4 Cipher Suites Supported (Bar Mitzvah)

#### Severity: Medium

#### Plug-in: #65821

**Port:** 21 / tcp / ftp

**Description:** 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.

**Solution:** 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.

**Business Impact:**

#### Vulnerability Name: SSL Certificate with Wrong Hostname

#### Severity: Medium

#### Plug-in: #45411

**Port:** 2078 / tcp / www

**Description:** The 'commonName' (CN) attribute of the SSL certificate presented for this service is for a different machine.

**Solution:** Purchase or generate a proper SSL certificate for this service.

#### Vulnerability Name: SSL Certificate Signed Using Weak Hashing Algorithm (Known CA)

#### Severity: Info

#### Plug-in: #95631

**Port:** 2078 / tcp / www

**Description:** The remote service uses a known CA certificate in the SSL certificate chain that has been signed using a cryptographically weak hashing algorithm (e.g., MD2, MD4, MD5, or SHA1). These signature algorithms are known to be vulnerable to collision attacks (CVE-2004-2761, for example). An attacker can exploit this to generate another certificate with the same digital signature, allowing the attacker to masquerade as the affected service.  
  
Note that this plugin reports all SSL certificate chains signed with SHA-1 that expire after January 1, 2017 as vulnerable. This is in accordance with Google's gradual sunsetting of the SHA-1 cryptographic hash algorithm.  
  
Note that this plugin will only fire on root certificates that are known certificate authorities as listed in Tenable Community Knowledge Article 000001752. That is what differentiates this plugin from plugin 35291, which will fire on any certificate, not just known certificate authority root certificates.  
  
Known certificate authority root certificates are inherently trusted and so any potential issues with the signature, including it being signed using a weak hashing algorithm, are not considered security issues.

**Solution:** Contact the Certificate Authority to have the certificate reissued.

#### Vulnerability Name: SSL Root Certification Authority Certificate Information

#### Severity: Info

#### Plug-in: #94761

**Port:** 2078 / tcp / www

**Description:** The remote service uses an SSL certificate chain that contains a self-signed root Certification Authority certificate at the top of the chain.  
  
**Solution:** Ensure that use of this root Certification Authority certificate complies with your organization's acceptable use and security policies.

#### Vulnerability Name: SSL/TLS Recommended Cipher Suites

#### Severity: Info

#### Plug-in: #156899

**Port:** 21 / tcp / ftp

**Description:** The remote host has open SSL/TLS ports which advertise discouraged cipher suites. It is recommended to only enable support for the following cipher suites:  
  
TLSv1.3:  
- 0x13,0x01 TLS13\_AES\_128\_GCM\_SHA256  
- 0x13,0x02 TLS13\_AES\_256\_GCM\_SHA384  
- 0x13,0x03 TLS13\_CHACHA20\_POLY1305\_SHA256  
  
TLSv1.2:  
- 0xC0,0x2B ECDHE-ECDSA-AES128-GCM-SHA256  
- 0xC0,0x2F ECDHE-RSA-AES128-GCM-SHA256  
- 0xC0,0x2C ECDHE-ECDSA-AES256-GCM-SHA384  
- 0xC0,0x30 ECDHE-RSA-AES256-GCM-SHA384  
- 0xCC,0xA9 ECDHE-ECDSA-CHACHA20-POLY1305  
- 0xCC,0xA8 ECDHE-RSA-CHACHA20-POLY1305  
- 0x00,0x9E DHE-RSA-AES128-GCM-SHA256  
- 0x00,0x9F DHE-RSA-AES256-GCM-SHA384  
  
This is the recommended configuration for the vast majority of services, as it is highly secure and compatible with nearly every client released in the last five (or more) years.

**Solution:** Only enable support for recommened cipher suites.

#### Vulnerability Name: HSTS Missing From HTTPS Server (RFC 6797)

#### Severity: Medium

#### Plug-in: #142960

**Port:** 443 / tcp / www

**Description:** The remote web server is not enforcing HSTS, as defined by RFC 6797. HSTS is an optional response header that can be configured on the server to instruct the browser to only communicate via HTTPS. The lack of HSTS allows downgrade attacks, SSL-stripping man-in-the-middle attacks, and weakens cookie-hijacking protections.

**Solution:** Configure the remote web server to use HSTS.