Six Degrees

IT Health Check – Penetration Testing
Service Description
Version 0.9
14 May 2021

1. Service Overview

The National Cyber Security Centre ("NCSC") IT Health Check ("ITHC") Penetration Testing Service provides you with an assessment of the level of risk to which your environment is exposed. The assessment identifies areas of weakness and provides a report that contains recommendations on how you can mitigate the risks raised during the test.

The ITHC is scoped for you to understand the security assurance of your environment. The ITHC includes multiple specific tests, and an output report with a summary of the number, type and severity of the issues identified and recommendations on how you can remediate these.

We are a member of the Check Scheme run by NCSC and are able to provide Government approved services. Additionally, our CHECK team members and leaders are certified under CREST, TigerScheme and Cyberscheme.

You can use or submit the report generated in this Service as an artefact to assist with the attainment of a recognised security accreditation or standard.

Additional information may be included in your Order Form or Statement of Work.

2. Service Deliverables

We offer the following types of penetration test services to conduct a NCSC IT Health Check:

- External Testing;
- Internal Testing;
- Build Review;
- Cloud Platform Build Review;
- System Configuration Review
- Mobile Application Testing;
- Web Application Testing; and
- Web Service / API Testing.

Not all service deliverables listed above will be required. We will agree with you and document the scope of testing specific to your environment in a Terms of Reference ("TOR") Document.

We will provide the necessary components to perform the checks according to your requirements as stated in your SoW. Your included services and the related fees will be specified in your Order Form.

If required, our Offensive Security Department can assist you with any prerequisite or supporting element, including the provision of IP information, credentials as required, provision of system configurations and provision of information using secure channels. If you require additional assistance to prepare for the test and to meet the dependencies listed in Section 6, we can deliver this support as part of a separate professional services engagement which will be subject to a separate fee as indicated on your Order Form.

2.1 External and Internal Testing

We offer:

(a) External penetration test to assess the level of risk that you and your users are exposed to over the internet ("External Testing"); and

(b) An internal penetration test to assess the level of risk that you and your users are exposed to from an attacker placed within your internal network ("Internal Testing").

Your requirements will be specified in your Order Form.

We will perform the penetration test service using one or more of the following tools (as deemed appropriate by us for your environment):

- Port scanner (nmap);
- Vulnerability scanner (Nessus);
- SSL/TLS scanner (testssl.sh);
- Text-based SSL/TLS client (openssl);
- Text-based network socket client (netcat);
- Service client (telnet, ftp); and
- Exploits and exploitation frameworks (service dependent).

The choice of tools will vary according to your requirements and your host environment. If we agree with you regarding specific tools for use, they will be specified in your SoW.

Testing Process

The testing process will be as follows:

(a) Port Scanning (Phase 1) – We will scan your target hosts and will enumerate the services that are exposed to the internet. We will inspect these open services to determine the purpose and the version details of any running software. These checks include but are not limited to identifying whether the web server software is out of date, if default or weak credentials can be used to authenticate to the site and determining if the SSL/TLS configuration can be improved.

(b) Vulnerability Scanning (Phase 2) – We will scan each open port via an automated scanner to identify potential vulnerabilities. We will analyse the open services information obtained through the scanning to check whether they match any profile of known vulnerabilities. Probes designed to elicit responses which can indicate the presence or absence of relevant vulnerabilities are sent to the target hosts and their open services.

We will exploit the vulnerabilities identified during the scanning and testing process in order to rule out false positive(s). Where multiple vulnerabilities are identified, we will exploit them in conjunction to build an attack chain.

(c) Penetration Testing (Phase 3) - We will probe each service manually to rule out any false positives and put together a potential attack tree. We will prioritise any vulnerability identified during the vulnerability scanning and we will manually examine the services to identify other potential vulnerabilities.

(d) Segregation Testing (Phase 4) – performed as part of our Internal Testing service only and not performed for External Testing. We will ensure that a destination network (A) which should be segregated from a source network (B), is in fact segregated and that a host in network (B) cannot access services which are in network (A). This phase is essentially a repeat of testing phase 1, but our testers are connected to each source network in turn and given a specific set of destinations to target from each source network.

Reporting

Reporting will include:

(e) Categorisation of each vulnerability;

(f) Threat exposure;

(g) Root cause of the issue;

(h) Testing technique used to find/reproduce the issue;

(i) Remediation advice; and

(j) Severity ratings.

We will determine the exploitability of each vulnerability identified during the penetration testing to provide you with an indication of what an attacker can achieve and help you to determine the extent of the risk to your organisation.

Please note that verification of vulnerabilities is not always possible and we will not attempt exploitation where this would pose an undue risk to the availability or integrity of your production systems and/or data. In these cases, we will categorise the severity of vulnerabilities according to the information available to the test team during the engagement and, therefore, there will be a higher chance of the vulnerability being a false positive.

We will report to you all the vulnerabilities categorised as critical.

2.2 Build Review

The Build Review assessment will determine how your in-scope systems have been configured to identify areas of weakness. We will investigate weaknesses and report on vulnerabilities with mitigation options, to offer feedback to harden your system configuration. We will conduct all reviews against the industries CIS benchmarking standard of configuration where applicable.

Build reviews can be carried out on servers, laptops, mobile phones, tablets, desktops and thin clients with core areas of examination including Operating System patching, third party patching, user account and passwords, configured services, applications installed, remote access configuration, general security configuration, antivirus and malware protection, firewall configuration, BIOS configuration and disk encryption settings. Your requirements will be specified in your Terms of Reference.

The testing process will be as follows:

(a) We will identify target hosts and their OS (not the purpose of the test);

(b) Our consultant will connect to the target system over the most appropriate administrative protocol (e.g. SSH, SMB, RDP), using an account set up with administrator level privileges;

(c) Depending on the type of system, we will use automated tools such as Nessus, and/or custom written audit scripts, to determine the patch level of the host and how well hardened it is against industry standard benchmarks;

(d) We will then perform a manual investigation based around the use of the system and any unusual features identified throughout the review process;

(e) Our consultant will review both the automated and manual results, and identify risks based on these results but also on the use and context of the server.

Reporting will include:

- Categorisation of each vulnerability;
- Threat exposure;
- Root cause of the issue;
- Testing technique used to find/reproduce the issue;
- Remediation advice; and
- Severity ratings.

We will perform the build review service using one or more of the following tools (as we determine to be the most appropriate):

- Vulnerability scanner (Nessus);
- Manual examination; and
- Custom written audit and examination scripts.

2.3 Cloud Platform Build Review

The Cloud Platform Build Review is applicable where there are infrastructures hosted within cloud platforms such as AWS and Azure or other SaaS cloud platforms. The review provides an environment build review option that assesses the configuration of cloud environments against best practice benchmarks for security such as CIS benchmarks as well our own ‘best practise’ standards for producing secure, minimised risk hosted environments.

The testing process will be as follows:

(a) Security Roles and Access Controls – A penetration tester lead review of the environments group policy settings to assure best practices are followed for the various roles and controls within the organisation. Our tester will also review access controls to assure appropriately strong protective policies are in place and applied correctly.

(b) Identity and Access Management – We will review the environments identity and access management configuration to ensure that access to said configuration tools are appropriate, and all users, power users and administrators are correctly configured assuring that privileged accounts are correctly utilised, and systems cannot be accessed inappropriately.

(c) Data Collection and Storage – We will perform manual and automated review of storage facilities in the cloud environment. We will review controls and encryption levels relating to data storage to assure encryption is correctly applied with a sufficient strength and that data in transit and storage is handled securely in line with best practice guidelines.

(d) Security Policies and Recommendations – We will review templates used and policies set for the deployment and build of systems to assure best practice is followed and that new builds are secure. We will assess controls around build and deployment and make recommendations as appropriate.

(e) Security Monitoring – We will review cloud security monitoring tools that are available and /or configured to assure key security incidents are alerted upon. Where tools are not used, we may make recommendations on key tool utilisation and configuration.

(f) Platform Breach Detection Capabilities – Similarly to Security monitoring, where tools are available to detect breach and compromise, we will review these to assure they have been set up correctly and provide visibility to the most common breach types, as well as being able to detect more advanced threats. Where tools are not utilised or are poorly configured, we will make key recommendations in line with best practice.

(g) Token, Keys and Keypair Configuration – Where additional levels of authentication and access controls are leveraged by token, certificates, keys or keypairs, we will undertake a review of the methods in place to check whether they are susceptible to attack or compromise. Where key lengths or technical controls are insufficient, or not meeting best practise, we will highlight these and make recommendations to strengthen or replace such controls.

(h) Load Balancing Configuration – Where applicable, we will undertake a review of the environment's load balancing configuration to assure best practice has been followed and that the configuration ensures equilibrium across devices. We will make recommendations as appropriate regarding how you can correct and optimise settings to prevent your system from being over utilised or susceptible to failure through overload.

(i) Application Resilience – We will review key applications to assure availability of systems. Where high availability services are enabled, we review their configuration to assess whether it allows for smooth system failover in the event of an issue and where application resilience is not utilised. Where appropriate, we may make recommendations to assist with your system availability.

Reporting will include:

- Categorisation of each vulnerability;
- Threat exposure;
- Root cause of the issue;
- Testing technique used to find/reproduce the issue;
- Remediation advice; and
- Severity ratings.

We will perform the build review service using one or more of the following tools (as we determine to be the most appropriate):

- Vulnerability scanner;
- Manual examination; and
- Custom written audit and examination scripts.

2.4 System Configuration Review

The System Configuration Review will determine how in scope network devices configuration has been implemented in order to identify areas of weakness. We will investigate the weaknesses and report on vulnerabilities with mitigation options, to allow a better hardened system configuration. We will conduct all reviews against the industries CIS benchmarking standard of configuration where applicable.

System Configuration reviews can be carried out on switches, routers, Wi-Fi controllers and firewalls. Core areas of examination include checks for whether any device rules allow unauthorised traffic in or out of specified zones/VLANs, whether the devices run insecure administration services (e.g. Telnet, SNMP V1), and the security of any device credentials stored locally as well as an analysis of all user accounts and their effective permission levels. During the review, we will check firmware version numbers and patching levels for the individual network devices to check whether the latest version of all software is being run, to identify vulnerabilities relating to any older versions identified, and to understand the risk the devices pose to the network infrastructure.

System Configuration Review is a non-intrusive service as all testing is carried out on the supplied configurations only. Testing will not be performed directly against the hosts.

The testing process will be as follows:

(a) We will use a combination of automated tools and manual configuration review to identify weaknesses within the device’s configuration;

(b) We will conduct checks for any known vulnerabilities against the current device firmware;

(c) We will conduct checks for missing patches;

(d) Where applicable, we will review all access control rules for security best practice;

(e) Where applicable, we will review all firewall rules for security best practice;

(f) Where applicable, we will review all device users and their appropriate security groups;

(g) We will check all device administration interface settings for security best practice.

Reporting will include:

- Categorisation of each vulnerability;
- Threat exposure;
- Root cause of the issue;
- Testing technique used to find/reproduce the issue;
- Remediation advice; and
- Severity ratings.

We will perform the build review service using one or more of the following tools (as we determine to be the most appropriate):

- Vulnerability scanner (Nessus);
- Manual testing; and

- Nipper.

2.5 Mobile Application Testing

The Mobile Application Testing Service consists of a security assessment of the mobile application based on OWASP principles and guidelines. This test focuses on the mobile application bespoke set of interactions with mobile devices.

The testing process will be as follows:

(a) Application Mapping (Phase 1) – We will examine the application to determine the extent of its standard features and functionality. One of our testers will map the application as an unauthenticated user and as each user role. The objective of this phase is to gain an understanding of the application prior the main testing.

(b) Security Testing (Phase 2) – We will perform a test of the categories listed in the OWASP mobile application methodology:

(i) Architecture, design and threat modelling;

(ii) Data storage and privacy;

(iii) Cryptography verification;

(iv) Authentication and session management;

(v) Network communication;

(vi) Platform interaction;

(vii) Code quality and build settings; and

(viii) Resiliency against reverse engineering.

The methodology for each category is customised to the specific mobile device platforms for which the application has been written.

Reporting will include:

- Categorisation of each vulnerability;
- Threat exposure;
- Root cause of the issue;
- Testing technique used to find/reproduce the issue;
- Remediation advice; and
- Severity ratings.

We will perform the mobile application test service using one or more of the following tools (as we determine to be the most appropriate):

- Mobile devices in their default and jailbroken states;
- Platform specific mobile application testing framework (Drozer, Needle, Frida, MobSF);
- Reverse engineering, including disassembly/reassembly (Apktool, Smali/Backsmali, Radare, IDA Pro, Hopper, cycript, APK studio);
- Web service/API security scanning and manual testing (Soap UI, Burp);
- Vulnerability-specific web scanner (sqlmap);
- SSL/TLS scanner (testssl.sh);
- Intercepting proxy (Burp);
- Text-based SSL/TLS client (openssl);
- Text-based network socket client (netcat);
- File metadata extractor (exiftool);
- Exploits and exploitation frameworks (application dependent); and
- Web browser (Chromium/Firefox).

The choice of tools will vary according to your requirements and your application. Any specific tolls agreed with you will be indicated in your SoW.

2.6 Web Application Testing

The Mobile Application Testing Service consists of a security assessment of the web application based on OWASP principles and guidelines. This test focuses on the application bespoke set of interactions with the web browser.

The testing process will be as follows:

(a) Application Mapping (Phase 1) – We will examine the application to determine the extent of its standard features and functionality. One of our testers will map the application as an unauthenticated user and as each user role. The objective of this phase is to gain an understanding of the application prior the main testing.

(b) Security Testing (Phase 2) – We will perform bespoke testing on each page, form, workflows and all of their individual parameters mapped during phase 1. We will test the categories listed in the OWASP mobile application methodology:

(i) Information gathering;

(ii) Service configuration;

(iii) Cryptography (Transport);

(iv) Identity and authentication;

(v) Authorisation and session management;

(vi) Input validation;

(vii) Error handling;

(viii) Cryptography (Application);

(ix) Application and business logic; and

(x) Client-side controls.

Where applicable, we will conduct the test from all user roles, including unauthenticated.

Reporting will include:

- Categorisation of each vulnerability;
- Threat exposure;
- Root cause of the issue;
- Testing technique used to find/reproduce the issue;
- Remediation advice; and
- Severity ratings.

We will perform the application test service using one or more of the following tools (as we determine to be the most appropriate):

- Web browser (Chromium/Firefox);
- Intercepting proxy (Burp);
- Web application vulnerability scanner (Burp and plugins);
- Vulnerability-specific scanner (sqlmap);
- SSL/TLS scanner (testssl.sh);
- Text-based SSL/TLS client (openssl);
- Text-based network socket client (netcat);
- File metadata extractor (exiftool);
- Web application framework scanner (wpscan); and

- Exploits and exploitation frameworks (application dependent).

The choice of tools will vary according to your requirements and your application. Any specific tolls agreed with you will be indicated in your SoW.

2.7 Web Service / API Testing

The Web Service/API Testing Service consists of a security assessment of the web service/API based on OWASP principles and guidelines. This test focuses on the application bespoke set of functionalities of the web service/API.

The testing process will be as follows:

(a) Web Service/API Workflow Mapping (Phase 1) – We will examine the web service/API to determine the extent of its standard features and functionality. One of our testers will map the web service/API as an unauthenticated user and as each user role. While doing so the tester will also take into consideration the application workflow (e.g. sequence of requests and responses that must be used to submit a valid subsequent request). The objective of this phase is to gain an understanding of the web service/API prior to the main testing.

(b) Security Testing (Phase 2) – We will perform bespoke testing on each request(s), method(s) and workflow(s) and all of their individual parameters mapped during phase 1. Where applicable, we will use all user roles, including unauthenticated, to attempt the access and/or subversion of requests, methods and workflows, in order to identify vulnerabilities. We will test the following categories:

(i) Broken object level authorisation;

(ii) Broken user authentication;

(iii) Excessive data exposure;

(iv) Lack of resources and rate limiting;

(v) Broken function level authorisation;

(vi) Mass assignment;

(vii) Security misconfiguration;

(viii) Injection;

(ix) Improper assets management; and

(x) Insufficient logging and monitoring.

Where applicable, we will conduct the test from all user roles, including unauthenticated.

Reporting will include:

- Categorisation of each vulnerability;
- Threat exposure;
- Root cause of the issue;
- Testing technique used to find/reproduce the issue;
- Remediation advice; and
- Severity ratings.

We will perform the test service using one or more of the following tools (as we determine to be the most appropriate):

- Request repeater (Burp);
- API parser/request generator (Soap UI, OpenAPI Parser);
- Web service vulnerability scanner (Burp and plugins);
- Vulnerability-specific scanner (sqlmap);
- SSL/TLS scanner (testssl.sh);
- Text-based SSL/TLS client (openssl);
- Text-based network socket client (netcat);
- File metadata extractor (exiftool);
- Exploits and exploitation frameworks (web service/API dependent); and
- Where appropriate, the web service/API dependent client - e.g. web application, mobile application, thick client.

The choice of tools will vary according to your requirements and your application. Any specific tolls agreed with you will be indicated in your SoW.

3. Service Delivery

After order acceptance, we will deliver the Service as follows:

Service Implementation	Us	You
We will discuss with you the Service and your environment to ensure accurate scoping for the penetration test performance.	●	●
You will provide us details of any required limitation around the scope of the penetration test(s).		●
You will provide us all the information necessary to conduct the penetration test(s).		●

Project approval and sign-off.		●
We will provide you with a Terms of Reference document confirming the scope and the schedule of the penetration test prior to commencement.	●	

All service delivery work will be carried out during Business Hours. We will invoice additional charges for any work undertaken out of hours. We reserve the right to charge additional fees for any change in your requirements occurring after the implementation of the Service and outside the scope of the contract (as described in the MSA).

3.1 Ready for Service

The Ready for Service Date is the date when we (acting reasonably and properly) notify you that the implementation steps are complete and that we are ready to conduct the penetration test. This date may be specified to occur at a later date, if mutually agreed to be so.

If you consider that the Service is not ready for service on the date notified, you must notify us within two (2) weeks of our notification, after which the Ready for Service date deemed to have occurred and we will proceed with the penetration testing on the date agreed with you and documented in your Terms of Reference. Any issues of which you may later become aware and notify us about will not change the Ready for Service date and may require additional chargeable services to be resolved.

You can request to postpone the test up to eleven (11) Business Days before the agreed test date indicated in your Terms of Reference, in which event we will reschedule the test and agree a new date with you for no additional charge.

In the event that you are not ready for the test on the agreed date and you fail to submit a request to postpone the test date with at least eleven (11) Business Days' prior notice, you will be able to cancel the test subject to a cancellation fee calculated as follows:

(a) A charge equal to 50% of the Non-Recurring Fee if you submit a cancellation request between eleven (11) and six (6) business days before the agreed test date; or

(b) A charge equal to 100% of the Non-Recurring Fee if you submit a cancellation request less than (6) business days before the agreed test date.

For the avoidance of doubt, any test that you wish to reschedule after cancellation shall be deemed a new order subject to a separate charge.

4. Billing

The Service will be billed as a professional service and Non-Recurring Fee as described in the Billing Guide, using the definition of the Ready for Service Date in Section 3.1 of this Service Description, and as indicated in your Order Form.

4.1 Additional Charges

We reserve the right to charge additional fees in the following situations:

(a) Implementation work carried out outside Business Hours; and

(b) Any change in your requirements occurring after the implementation of the Service and outside the scope of the contract (as described in the MSA).

5. Service Operations

Please refer to our Operations Manual for further information on incident management, requests for change and information requests.

6. Dependencies - External Testing

We provide External Testing subject to the following dependencies:

6.1 Client Obligations

(a) You will provide us details of your target hosts and any required information around scope (e.g. list of IPs or ports on which performing the testing);

(b) You will obtain authorisation to test from network/hosting providers, or confirmation of a non-requirement for authorisation in the case of providers such as Microsoft Azure; and

(c) You will ensure all our IP addresses are applied to any border firewalls or other whitelists as required.

6.2 Prerequisites

(a) You will provide signed acceptance and approval of the project before testing commences.

6.3 Service Dependencies

(a) Before commencing the service, we will provide you with our source IP addresses and you will disable any DDoS protection or other network controls that could hinder testing for our source IP;

(b) If web applications are present, you will provide us the relevant host name to allow access; and

(c) If applicable, you will provide us details and/or firewall rules set to allow bypass of third-party DDoS/WAF protection.

7. Dependencies - Internal Testing

We will provide the Internal Testing subject to the following dependencies:

7.1 Client Obligations

(a) You will provide us details of your target hosts and any required information around scope (e.g. list of IPs or ports on which performing the testing);

(b) You will provide us your source IP address, network mask, gateway and any relevant VLAN details for the network segments from which the tests should be conducted from; and

(c) You will ensure that the source IP addresses you provide us are applied to any firewall, router/switch access controls list or other whitelists as required to facilitate access to the target subnets.

7.2 Prerequisites

(a) You will provide signed acceptance and approval of the project before testing commences.

7.3 Dependencies

(a) If web applications are present, you will provide us the relevant host name to allow access;

(b) If applicable, you will provide us details and/or firewall rules set to allow bypass of any intrusion prevention or other network filtering systems; and

(c) To allow the performance of segregation testing, you will provide us a list of source networks which the tester will be placed in, and a list of destination networks or hosts which the tester should target from each source network.

8. Dependencies - Build Review

We will provide the Build Review/s subject to the following dependencies:

8.1 Client Obligations

(a) You will provide us details of your target hosts and any required information around scope (e.g. list of IPs and associated Operating System on which we will be performing the review);

(b) You will provide us local administrator and/or root level user accounts for the systems being reviewed.

(c) You will provide us your source IP address, network mask, gateway and any relevant VLAN details for the network segments from which the tests should be conducted from; and

(d) You will ensure that the source IP addresses you provide us are applied to any firewall, router/switch access controls list or other whitelists as required to facilitate access to the target subnets.

8.2 Prerequisites

(a) You will provide signed acceptance and approval of the project before testing commences.

8.3 Dependencies

(a) Access and authentication as requested must be available for the review to be completed; and

(b) If applicable, you will provide us details and/or firewall rules set to allow bypass of any intrusion prevention or other network filtering systems.

9. Dependencies - Cloud Platform Build Review

We will provide the Cloud Platform Build Review/s subject to the following dependencies:

9.1 Client Obligations

(a) You will provide us details of your target platforms and any required information around scope (e.g. list of URLs on which we will be performing the review);

(b) You will provide us a read only administrator and/or root level user accounts for the systems being reviewed;

(c) You will provide us your source IP address, network mask, gateway and any relevant VLAN details for the network segments from which the tests should be conducted from; and

(d) You will ensure that the source IP addresses you provide us are applied to any firewall, router/switch access controls list or other whitelists as required to facilitate access to the target subnets.

9.2 Prerequisites

(a) You will provide signed acceptance and approval of the project before testing commences.

9.3 Dependencies

(a) Access and authentication as requested must be available for the review to be completed; and

(b) If applicable, you will provide us details and/or firewall rules set to allow bypass of any intrusion prevention or other network filtering systems.

10. Dependencies - System Configuration Review

We will provide the System Configuration Review subject to the following dependencies:

10.1 Client Obligations

(a) You will ensure all device configurations are provided in a plain text human readable format. Common examples are: XML, JSON, HTML, output from “show running config” on Cisco and like devices are all acceptable;

(b) You will take note of how many configurations have been quoted for and ensure only this number of configurations have been submitted for review; failure to do so may result in submitted configurations being rejected;

(c) You will ensure paired devices are both submitted where applicable;

(d) You will ensure you are sending the device configurations to us in a secure manner (encrypted); and

(e) For firewall reviews in particular, you will provide us accurate information on context in terms of network diagrams and network context. Please note that the accuracy of the report is subject to the amount of information provided by you.

10.2 Prerequisites

(a) You will provide signed acceptance and approval of the project before testing commences.

10.3 Dependencies

(a) Access and authentication as requested must be available for the review to be completed; and

(b) If applicable, you will provide us details and/or firewall rules set to allow bypass of any intrusion prevention or other network filtering systems.

11. Dependencies - Mobile Application Testing

We will provide the Mobile Application Testing subject to the following dependencies:

11.1 Client Obligations

(a) You will provide us mobile application binaries, download URLs, or platform marketplace details;

(b) You will provide us backend web service/API URL(s) and additional relevant details, including any required limitation around scope (e.g. specific endpoints or HTTPS);

(c) You will provide us the credentials for all user roles to be tested where authenticated testing applies, including documentation of the privileges assigned to and relationships between the user roles where multiple user roles are being tested;

(d) You will obtain authorisation to test from network/hosting providers, or confirmation of a non-requirement for authorisation in the case of providers such as Microsoft Azure; and

(e) You will ensure all out IP addresses are applied to any border firewalls or other whitelists as required.

11.2 Prerequisites

(a) You will provide signed acceptance and approval of the project before testing commences.

11.3 Dependencies

(a) Before commencing the service, we will provide you with our source IP addresses and you will disable any relevant web application firewalls for our source IP;

(b) If applicable, you will provide us details and/or firewall rules set to allow bypass of third-party DDoS/WAF protection; and

(c) Where multiple IP addresses are resolved from the URL hostname, we will focus the testing on one IP address.

12. Dependencies - Web Application Testing

We will provide the Web Application Testing subject to the following dependencies:

12.1 Client Obligations

(a) You will provide us target URL, full web service/API documentation and additional relevant details, including any required limitation around scope (e.g. specific endpoints/methods or HTTPS);

(b) You will provide us the credentials for all authorisation roles to be tested where authenticated testing applies, including documentation of the privileges assigned to and relationships between the authorisation roles where multiple user roles are being tested;

(c) You will obtain authorisation to test from network/hosting providers, or confirmation of a non-requirement for authorisation in the case of providers such as Microsoft Azure; and

(d) You will ensure all our IP addresses are applied to any border firewalls or other whitelists as required.

12.2 Prerequisites

(a) You will provide signed acceptance and approval of the project before testing commences.

12.3 Dependencies

(a) Before commencing the service, we will provide you with our source IP addresses and you will disable any relevant web application firewalls for our source IP;

(b) If applicable, you will provide us details and/or firewall rules set to allow bypass of third-party DDoS/WAF protection;

(c) If applicable, you will provide us details of any web service/API dependent client device and/or software; and

(d) Where multiple IP addresses are resolved from the URL hostname, we will focus the testing on one IP address.

13. Dependencies – Web Service/API Testing

13.1 Client Obligations

(a) You will provide us target URL, full web service/API documentation and additional relevant details, including:

(i) Limitations around scope, including if testing should be limited to particular endpoints, if particular methods should be used or avoided, if testing should only be carried out on HTTPS and not HTTP or any other consideration;

(ii) Provide all IP addresses that are resolved from the URL hostname (Our Test team will choose one IP and testing will be carried out against this);

(iii) Where appropriate, details should be provided and/or firewall rules set to allow bypass of third-party DDoS/WAF protection. Although we do not conduct DDoS tests, these systems can hinder testing in a way that would not affect an attacker that is not confined to the limitations of engagement time-frames.

(b) You will obtain authorisation to test from network/hosting providers, or confirmation of a non-requirement for authorisation in the case of providers such as Microsoft Azure; and

(c) You will provide authorisation, or where appropriate, obtain authorisation from the Service/API owner that web application firewalls and security controls can be disabled and/or amended as required.

13.2 Prerequisites

(a) You will provide signed acceptance and approval of the project before testing commences.

13.3 Dependencies

(a) Before commencing the service, we will provide you with our source IP addresses and you will disable any relevant web application firewalls for our source IP, you will ensure all our IP addresses are applied to any border firewalls or other whitelists as required;

(b) You will provide us the credentials for all authorisation roles to be tested where authenticated testing applies, including documentation of the privileges assigned to and relationships between the authorisation roles where multiple user roles are being tested;

(c) If applicable, you will provide us details and/or firewall rules set to allow bypass of third-party DDoS/WAF protection and assure that any web application firewalls are disabled for the test source IP address;

(d) Where appropriate, you will provide web service/API dependent client device/software, e.g. web application, mobile application or thick client software. This allows greater understanding of the web service/API and can allow the full business risk of certain types of vulnerability to be determined.

14. Exclusions

Please note that verification of vulnerabilities is not always possible and we will not attempt exploitation where this would pose any undue risk to the availability or integrity of production systems and/or data.

The excluded items described below are outside the scope of this Service Description:

14.1 General exclusions:

(a) Any retesting following agreed test period unless specified in your SoW;

(b) Completion of any remediation actions; and

(c) Creation of risk treatment plans.

14.2 External Testing and Internal Testing

(a) Denial of service;

(b) Testing that may affect the availability or integrity of production systems or data; and

(c) Full web application assessment.

14.3 Build Review and Cloud Platform Build Review

(a) Denial of service;

(b) Data deletion;

(c) Data alteration;

(d) Power cycling of the host; and

(e) Any activity that would compromise the BAU use of the host.

14.4 System Configuration Review

(a) We will not be able to review unreadable or corrupt configurations.

14.5 Mobile Application Testing

(a) Denial of service;

(b) Testing that may affect the availability or integrity of production systems or data;

(c) Mobile device testing, multiple mobile application testing, web application testing, and web service/API testing.

14.6 Web Application Testing

(a) Denial of service;

(b) Testing that may affect the availability or integrity of production systems or data;

(c) External or internal network testing, multiple web services/API testing, mobile application testing, web application testing, and thick client application testing;

(d) Non-standard or unusual web services are excluded from the testing as we cannot ensure the correct functioning of usual testing tool on an API using a non-standard, unusual or proprietary protocol; and

(e) Other types of API testing (non-web services API).

14.7 Web Service/API Testing

(a) Denial of service;

(b) Testing that may affect the availability or integrity of production systems or data;

(c) External or internal network testing, multiple web services/API testing, mobile application testing, web application testing, and thick client application testing;

(d) Non-standard or unusual web services are excluded from the testing as we cannot ensure the correct functioning of usual testing tool on an API using a non-standard, unusual or proprietary protocol; and

(e) Other types of API testing (non-web services API).

15. Definitions and Acronyms

15.1 Definitions

In this document “we” or “us” refers to the Supplier, and “you” refers to the Client.

The terms listed have the following meanings:

Term	Meaning
BIOS	Basic firmware used to control underlying hardware and facilitate basic setup and control.
CHECK	CHECK is the term for the NCSC approved penetration test companies and the methodology used to conduct a penetration test.
Mobile Application	A piece of computer software which allows users to interact with servers on the internet or other network (e.g. corporate internal network) using a mobile device such as a phone or a tablet.
Terms of Reference	A document which defines project-specific or client-specific activities, deliverables and/or timelines for providing services to that client.

URL	Uniform Resource Locator; a web address.
Web Application	A piece of computer software which allows users to interact with servers on the internet or other network (e.g. a corporate internal network) using a web browser.
Web Service/API	An API provides the means by which different software components can talk to one another. A web service is a type of API which is exposed over a network such as the internet and uses web technologies such as web servers, HTTP and SSL/TLS to form the communications channel.

15.2 Acronyms

Acronym	Meaning
API	Application Program Interface
BAU	Business as Usual
BIOS	Basic Input/Output System
DoS	Denial of Service
ITHC	IT Health Check
NCSC	National Cyber Security Centre
OWASP	Open Web Application Security Project
RDP	Remote Desktop Protocol
SMB	Server Message Block protocol
SSH	Secure Shell protocol
SSL	Secure Sockets Layer
SoW	Statement of Work
TLS	Transport Layer Security protocol
ToR	Terms of Reference
URL	Uniform Resource Locator

16. Operational Targets

N/A