



Euclid Express Application Penetration Test – Technical Report

Euclid / Project Number: OP-106721

Revision Number: 2.0

Date: 10/8/2014

Table of Contents

Executive Summary.....	3
Scope.....	3
Overall Summary and Conclusions	4
<i>Identified Strengths</i>	4
<i>Key Findings</i>	4
Overview of Testing Results by Group	4
Application Profile	6
Environment and Architecture Summary	6
Authentication.....	7
<i>Authentication and Password Controls</i>	7
<i>Session Management Controls</i>	7
Authorization	7
Threat Analysis.....	8
Findings and Recommendations	9
Finding Composition	9
<i>Definition of a Technical Finding</i>	9
<i>Description of Findings Groups</i>	9
Findings Ranking System	10
<i>Severity Categories</i>	10
Findings Matrix.....	11
Application Assessment Findings.....	12
Application Architecture	12
<i>Third-party script inclusion</i>	12
<i>Lack of Login Bruteforce Protections</i>	12
<i>Concurrent Sessions Allowed</i>	13
Environment and Configuration	13
<i>Password field with Autocomplete enabled</i>	13
<i>Weak SSL Ciphers Supported</i>	14
Appendix A – Project Information.....	15
Assessment Project Team	15
Appendix B – Methodology Overview	17
Application Profiling	17
Threat Analysis	17
Dynamic Testing.....	17
Appendix C – Remote Testing Results	19

Executive Summary

Euclid engaged Accuvant LABS to perform a security assessment of the organization's Euclid Express application and supporting application environment. This report details the assessment of the application and supporting environment as of September 2014. The objective was to assess the current security posture and the effectiveness of the controls in place within the application environment, compare the results of the assessment with industry best practices and identify vulnerabilities that could negatively affect the application or business as a whole.

As an addition to the application assessment, Euclid requested that Accuvant perform remote network testing of public-facing systems. The purpose of this scan was to find common misconfigurations, open ports, outdated cryptographic protocols in use. The scope of systems tested included those servers that host the Euclid Express application, as well as other supporting systems. The results of that testing are included in [Appendix C](#), as well as the supplemental spreadsheet distributed with this report.

It is important to note that this report represents a snapshot of the security of the environment assessed at a point in time. Conditions may have improved, deteriorated or remained the same since this assessment was completed.

Accuvant LABS knows the importance Euclid places on data security and sincerely appreciates the opportunity to have worked with Euclid on this engagement. Should you have any questions regarding these findings or the contents of this report, please feel free to contact us.

Scope

Euclid identified the following application components for the application security assessment:

- Euclid Express web application

Euclid identified the following application components for the remote network security assessment:

- Euclid Express web application
- Euclid Express development and test environments
- Other hosts within the Euclid network space (190 total)

The following application components were out of scope for this engagement:

- Legacy systems and underlying infrastructure components
- Underlying shared service infrastructure components
- All other Euclid applications and systems

Project Scope Details

Project Scope Details	
Application Name and Version	Euclid Express 3.0
Engagement Length	11 days for application testing, Up to 8 hours for network-based and application-based injection point manual testing.
Application Access	Working instance of the application, source code, server configuration, and developer documentation
Consultant Location	Testing was performed remotely via Internet.
Environment	Production
User Roles	4 – 2 administrative users, 2 standard users

The phases and associated sub-components of this assessment included:

Assessment Phase	Tasks
Application Profiling	Through a review of available documentation, runtime analysis and developer interviews, the assessment team created a profile of the application security model and its key functionality.
Threat Analysis	The assessment team documented critical data held by the application and likely attacker goals based on input from the application owners and the assessment team's experience.
Dynamic Testing	The assessment team executed manual testing procedures, including unauthenticated and authenticated test scenarios within the user roles defined for the engagement. The application execution environment was also tested using a comprehensive suite of application and network assessment tools and manual verification, to identify common vulnerabilities. Where vulnerabilities were found, the assessment team created a proof of concept to demonstrate the issue and provided reproduction information for the development staff.
Remote Network Testing	The assessment team executed automated, unauthenticated tests using a variety of network assessment tools and manual verification, to identify common remote vulnerabilities.

Overall Summary and Conclusions

Based on the severity of the vulnerabilities discovered during the course of this engagement, the overall security posture of Euclid Express is at a low level of risk. Accuvant LABS based this conclusion on the following key findings:

Identified Strengths

- Web servers for the application are appropriately hardened and do not suffer from easily discovered or well-known vulnerabilities or information disclosures.
- Direct attacks against the application and environment did not yield access to sensitive data assets.
- The application has a small attack surface and uses best practice design patterns.
- No cross-site scripting (XSS) issues were discovered.
- No significant patching issues were discovered.




Key Findings

- A number of low severity findings such as weak session management practices, autocomplete, and third-party script inclusion were identified. These represent opportunities to improve the overall security of the application rather than serious defects.
- Some supported SSL cipher suites are considered to be weak and should be disabled if possible.
- Remote network testing found a large number of low-severity configuration weaknesses on a variety of hosts.

The findings section of this report details weaknesses where the application does not implement best practices for a particular security setting or control. These create information disclosures and increase the attack surface of the application. These issues are expected to be straightforward to address.

Overview of Testing Results by Group

The table below provides a high-level overview of the observations based on each group of findings. It highlights key findings by category and provides an associated summary ranking of severity based on susceptibility to exploitation, threat exposure and environment attack surface.

Findings Group	Severity	Key Observations
Application Architecture		<ul style="list-style-type: none"> • Third-party Script Inclusion – Content from an external source is allowed to execute in the context of the application. • Lack of Login Bruteforce Protections – No rate limiting or lockouts are enforced by the application.
		<ul style="list-style-type: none"> • Concurrent Sessions Allowed – Users may login to the same account simultaneously from multiple locations.
Environment & Configuration		<ul style="list-style-type: none"> • Password Autocomplete Enabled – Autocomplete is enabled on password fields in the application. • Weak SSL Ciphers Supported – Some SSL ciphers supported by the application are considered to be weak.

Application Profile

Euclid's products are used to collect data from brick and mortar locations about customer activity. This data can inform customer retention and engagement strategies. Euclid Express implements an interface for viewing, interacting with, and generating reports on data collected. Customizable dashboards and downloadable reports are included in the web application.

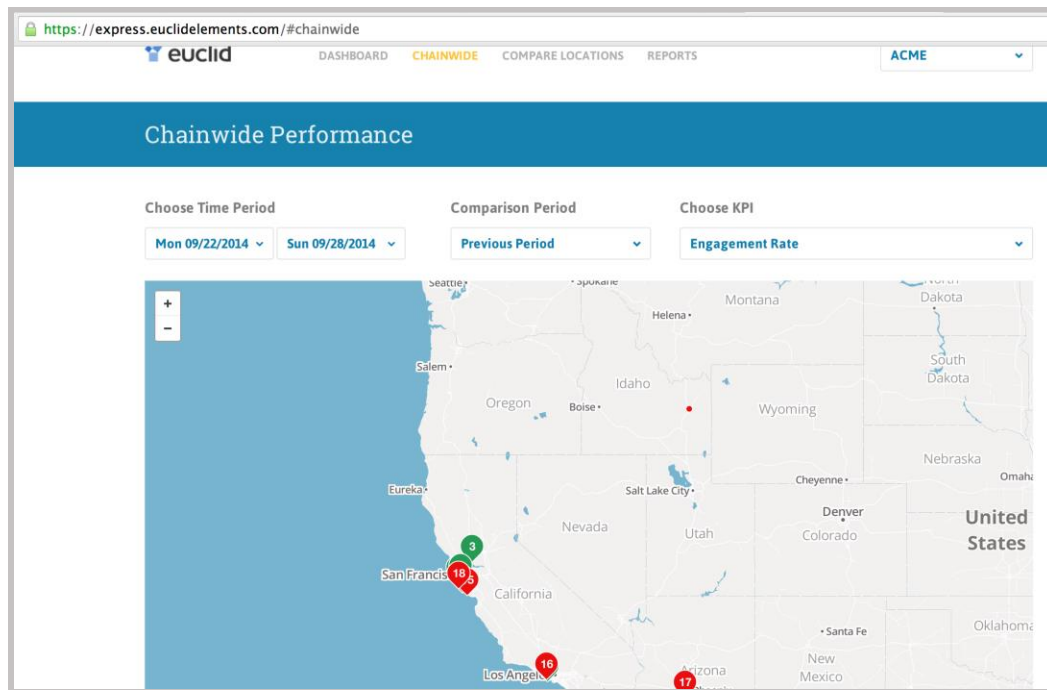


Figure 1 - A typical page from the Euclid Express application

Environment and Architecture Summary

The Euclid Express application is implemented in Ruby and runs on a Rails environment in the Amazon Cloud. The application has three layers: presentation, business logic and data access. The presentation layer is implemented in HTML and JavaScript, with the latter language used heavily with AJAX and JSON to create an interactive, intuitive user experience. Several third-party JavaScript libraries are in use, some of which, such as Google Analytics and Mapbox, are hosted on a third-party server.

Database services use MySQL Server. The application leverages third-party frameworks and libraries, including the JQuery framework as well as various Ruby libraries.

The table below provides additional details about the application and components.

Application Detail	
Application Type	Web Application with supporting Web Services
Languages in Use	Ruby, JavaScript, HTML, CSS
Frameworks & Components	Rails 4.1.0, aws-sdk, Google Analytics, Mapbox 2.0.1
Database Type	MySQL
Application Version	3.0
Revision Level	Release

Application Detail

Tested Entry Points	https://express.euclidelements.com
---------------------	---

Authentication

The application uses form-based authentication to authenticate users. Users provide a UserID and password as login credentials. Passwords can be changed after login. User authentication credentials are stored in the application database.

Authentication and Password Controls

- Passwords must be a minimum of eight characters.

Session Management Controls

- A session cookie token (_express_session) tracks and manages unique user sessions. The 'Secure' and 'HttpOnly' flags issue the session cookie in order to control against session hijacking.
- For Internet users, inactive sessions expire when the browser is closed.



https://express.euclidelements.com/users/sign_in	
▼ 1 cookie	
Name	_express_session
Value	WIZsTHVVMG4vbXhxbUNqMTZCS2twdnc1YjQrckQxam9iVDRLa3RmaVhTV3YyNWdnNXVhaU1TczFpcDBYQ0kwcD%3D%3D--1402e676722f42e348d714310e782e6bc0622e2a
Host	express.euclidelements.com
Path	/
Expires	At end of session
Secure	Yes
HttpOnly	Yes

Figure 2 - Euclid Express session material

Authorization

The table below lists the accounts created for this assessment and their access rights. The table below lists user accounts in relative order of access, with the highest privileged accounts listed first.

User Account	Role/Access Level	Description of Access
test_acme_admin1@euclidelements.com	Administrator	Ability to modify locations, import data, and use reporting functions.
test_acme_admin2@euclidelements.com	Administrator	Ability to modify locations, import data, and use reporting functions.
test_acme_user1@euclidelements.com	User	Lowest privilege level of access. Read-only access to reporting and data view functions.
test_acme_user2@euclidelements.com	User	Lowest privilege level of access. Read-only access to reporting and data view functions.

Threat Analysis

In this phase, Accuvant LABS created a basic threat analysis for the application. This includes critical data held by the application and likely attacker goals. This analysis was performed using input from the product owners, along with the experience of the assessment team. The table below describes the data or functionality that the Euclid Express application should protect. The items listed in the table would be the objective of a successful application compromise.

Name	Description
User credentials	Username, password hashes, etc.
Store data	Data collected by the Euclid Express system
Encryption Keys	Private keys and salts used to encrypt data stored in the database.

Based on input from the development group and product owners, the primary threats the product team is concerned with include:

- Compromise of store data via the web application interface.
- Modification of store data or disclosure to unauthorized business partners.
- Loss of encryption key material via an application attack.
- Hijacking user session.
- Cross-site scripting attacks to compromise user sessions.
- Abuse of authorization and automation controls on backend services.
- Insertion of malicious data into the database.

Findings and Recommendations

Because of this assessment, Accuvant LABS identified a number of areas where security controls could be improved, augmented or refined. The remainder of this report describes the details of Accuvant LABS' observations regarding security vulnerabilities and/or control deficiencies, the severity associated with the issues identified and recommendations for resolving those issues.

Accuvant LABS recommends that Euclid developers first test the recommended changes to ensure that they do not adversely affect application functionality.

Finding Composition

Definition of a Technical Finding

Technical findings in this document each represent a class of security vulnerabilities identified during testing. Findings are grouped based on common root causes. For example, if there were 10 unique URLs vulnerable to SQL injection in an application, a single SQL injection finding would be documented with general remediation steps. Within the finding, each instance of SQL injection vulnerability would be enumerated.

Description of Findings Groups

The findings groups are described as follows:

- **Application Technical Findings** – Application technical findings represent issues within the application that directly relate to a confirmed or potential attack vector. In these findings, the application implementation deviates from best practices for secure application design or development. Exploitation may result in violation of data integrity, application availability or data confidentiality.
- **Application Architecture Findings** – Application architecture findings relate to specific application design components that do not align with best practices. These issues are not specific vulnerabilities in the application, but can increase the attack surface, increase the likelihood of the presence of an attack vector or elevate the severity of existing attack vectors. The degree of deviation from appropriate controls and the potential impact to the application determines the severity level.
- **Environment and Configuration Findings** – Environment and configuration findings deal with issues that weaken the security posture of the application's supporting hosts and services. This includes findings related to the patch level of exposed hosts and services. This group also contains findings in system and service configurations that deviate from industry best practices and company policies. The existence of direct attack vectors for some issues in this category determines the severity level.

Findings Ranking System

In order to prioritize the assessment results, each finding was categorized based on severity classifications. Final analysis of the risk or impact to the application will require an internal evaluation by Euclid personnel. Accuvant LABS has developed classifications using the severity nomenclature for ranking the issues identified within the various severity categories.

Severity Categories

Based on Accuvant LABS' analysis of the particular finding and assets affected, a finding will fall into one of the following severity level categories:



Severity – Critical: Critical vulnerabilities require an immediate response through mitigating controls, direct remediation or a combination thereof. Exploitation of critical severity vulnerabilities results in privileged access to the target system, application or sensitive data and enables further access to other hosts or data stores within the environment. Findings with a critical ranking will cause significant losses when they are exploited, although the total cost is difficult to quantify in advance. In general, a critical severity ranking is warranted when the issue has a direct impact on regulatory or compliance controls imposed on the environment, accesses personally identifiable information (PII) or financial data or could cause significant reputational or financial harm.



Severity – High: Findings with a high severity ranking require immediate evaluation and subsequent resolution. Exploitation of high severity vulnerabilities leads directly to an attacker gaining privileged, administrative-level access to the system, application or sensitive data. However, it does not enable further access to other hosts or data stores within the environment. If left unmitigated, high severity vulnerabilities can pose an elevated threat that could affect business continuity or cause significant financial loss.



Severity – Medium: A finding with a medium severity ranking requires review and resolution within a short time. From a technical perspective, vulnerabilities that warrant a medium severity ranking can lead directly to an attacker gaining non-privileged or user-level access to the system, application or sensitive data. Findings that can cause a denial-of-service (DoS) condition on the host, service or application are also classified as medium risk. Alternately, the vulnerability may provide a way for attackers to gain elevated levels of privilege. From a less technical perspective, observations with this ranking are significant, but they do not pose as much of a threat as high or critical severity exposures.








Severity – Low: Low severity findings should be evaluated for review and resolution once the remediation efforts for critical, high and medium severity issues are complete. From a technical perspective, vulnerabilities that warrant a low severity ranking may leak information to unauthorized or anonymous users used to launch a more targeted attack against the environment. From a process perspective, observations with this ranking provide awareness and should be addressed over time as part of a comprehensive information security program, but do not presently pose a substantial threat to business operations or have any significant loss associated with the exposure.



Informational: An informational finding presents no direct threat to the confidentiality, integrity or availability of the data or systems supporting the environment. These issues pose an inherently low threat to the organization and any proposed resolution should be considered as an addition to the information security procedures already in place.

Findings Matrix

The table below provides a summary of the assessment findings categorized by group and ranked by severity. The table provides Euclid with an overview of all of the findings from the assessment and allows the remediation team to focus efforts on the areas of highest severity as determined by Accuvant LABS. Click the individual link below to go directly to that finding.

Finding Category	Severity
Application Architecture	
Third-Party Script Inclusion	
Lack of Login Bruteforce Protections	
Concurrent Sessions Allowed	
Environment and Configuration	
Password Field With Autocomplete Enabled	
Weak SSL Ciphers Supported	

Application Assessment Findings

Application Architecture

Application architecture findings relate to specific application design components that do not align with best practices. These issues are not specific vulnerabilities in the application, but can increase the attack surface, increase the likelihood of the presence of an attack vector or elevate the risk level of existing attack vectors. The degree of deviation from appropriate controls and the potential impact to the application determines the severity level.



Third-party script inclusion

Observation: The application includes remote code from api.tiles.mapbox.com, a third-party server, via a <SCRIPT> tag. When an application includes a script from an external domain, this script is executed by the browser within the security context of the invoking application. The script can therefore do anything that the application's own scripts can do, such as accessing application data and performing actions within the context of the current user.

When executable code is included from an external domain, that domain is being trusted with the data and functionality of the application. Innocent or malicious changes to the remote code can both be effective mechanisms for indirectly attacking the Euclid Express application.

Reproduction: View the HTML source on any page of the application. Note that remote code, at [https://api.tiles.mapbox.com/mapbox.js/v2.0.1 /mapbox.js](https://api.tiles.mapbox.com/mapbox.js/v2.0.1/mapbox.js), is included via a <SCRIPT> tag on line 12 of the HTML header.

Severity – Low: While a third-party server may be considered “trusted,” including any remote content, in particular executable code, from a foreign server makes it impossible to guarantee the integrity and security of an application.

The strength and integrity of the application is, at best, reliant on the strength and integrity of the third party system.

Recommendation: Host all executable content on a local, trusted system. If remote content must be included, implementing measures to monitor that content for changes may assist in ensuring application integrity, but will not prevent potential upstream trust issues such as DNS hijacking, DNS pinning, or other network redirection attacks on the client side.

Asset(s) Affected:

- <https://express.euclidelements.com/>



Lack of Login Bruteforce Protections

Observation: No server-side countermeasures are in place to limit application login attempts. Weak login credentials may be discovered through automated brute force or dictionary-style attacks.

Reproduction: Attempt to login many times with invalid credentials. No limit on the number of attempts is enforced.

Severity – Low: An attacker with knowledge of legitimate usernames or username-generation schemas may discover accounts with weak passwords. Dictionaries and cracking software are widely available for automating bruteforce-style attacks, making them efficient and popular methods for attackers.

Recommendation: Failed login attempts to a single account should be logged on the server side and limited to a reasonable number, at which point the account should be locked and require administrative approval or positive confirmation of user's identity before restoring access. Tracking and limiting failed logins by IP address or other remote identifier is unreliable, as those identifiers can be controlled by the remote user, however this may be a useful technique for limiting the "noise" of failed attempts.

Require and enforce strong user passwords on account creation and password change.

Asset(s) Affected:

- https://express.euclidelements.com/users/sign_in



Concurrent Sessions Allowed

Observation: A single user can maintain multiple simultaneous sessions on the same or multiple browser sessions.

Reproduction: Log in to the application with the same account in multiple browsers. Note that both sessions remain valid.

Severity – Informational: Unless the use case for the application specifically requires it, concurrent logins should be discouraged. The ability to log in from multiple browsers increases the application's level of exposure to malicious and accidental abuse, including direct abuse of session management functions and shared credentials between staff and other end-users.

Recommendation: Modify application logic on a successful login to determine whether a session exists for that user, provide an informative logout message to both sessions, and discontinue the previously existing session.

Asset(s) Affected:

- https://express.euclidelements.com/users/sign_in

Environment and Configuration

Environment and configuration findings deal with issues that weaken the security posture of the application's supporting hosts and services. This includes findings related to the patch level of exposed hosts and services. This group also contains findings in system and service configurations that deviate from industry best practices and company policies. The existence of direct attack vectors for some issues in this category determines the severity level.



Password field with Autocomplete enabled

Observation: Most browsers have a facility to remember user credentials that are entered into HTML forms. This function can be configured by the user and also by applications which employ user credentials. If the function is enabled, then credentials entered by the user are stored on their local computer and retrieved by the browser on future visits to the same application.

The stored credentials can be captured by an attacker who gains access to the computer, either locally or through remote compromise. Further, methods have existed whereby a malicious web site can retrieve the stored credentials for other applications, by exploiting browser vulnerabilities or through application-level cross-domain attacks.

Reproduction: View the HTML source on the application's "Sign In" page. Note that neither the <FORM> tag nor the password's <INPUT> tag contains the autocomplete="off" HTML attribute.

Severity – Informational: Autocomplete issues are rarely exploitable by themselves, but such features can often be leveraged by secondary vulnerabilities to reveal sensitive data about the user.

Recommendation: To prevent browsers from storing credentials entered into HTML forms, include the attribute autocomplete="off" within the FORM tag (to protect all form fields) or within the relevant INPUT tags (to protect specific individual fields).

Asset(s) Affected:

https://express.euclidelements.com/users/sign_in



Weak SSL Ciphers Supported

Observation: The remote host supports the use of SSL ciphers that operate in Cipher Block Chaining (CBC) mode. The remote host also supports use of RC4 ciphers. Both these cryptographic technologies have been shown to contain weaknesses in specific situations and are not recommended for production systems.

Severity – Informational: CBC cipher suites offer additional security over Electronic Codebook (ECB) mode, but have the potential to leak information if used improperly. The RC4 algorithm, as used in the TLS protocol and SSL protocol, has many single-byte biases, which makes it easier for remote attackers to conduct plaintext-recovery attacks via statistical analysis of ciphertext in a large number of sessions that use the same plaintext.

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

Asset(s) Affected:

- <https://express.euclidelements.com>

Appendix A – Project Information

This section provides an overview of the Accuvant team members assigned to the project.

Assessment Project Team

The tables below contain a listing of the Accuvant personnel involved in this engagement. Feel free to contact the appropriate persons at any time to discuss the results of the engagement.

Name:	Mike Bailey
Title:	Senior Software Security Consultant – Accuvant LABS
Phone:	Mobile: +1.801.380.3442
E-mail Address:	mbailey@accuvant.com
Project Involvement:	Served as the project lead and performed technical testing and analysis. Direct questions regarding the deliverable here.

Name:	Robert Kornmeyer
Title:	Consultant – Accuvant LABS
Phone:	Mobile: +1.315.404.1611
E-mail Address:	rkornmeyer@accuvant.com
Project Involvement:	Performed network-based testing

Name:	John Bock
Title:	Director –Software Security Group – Accuvant LABS
Phone:	Mobile: +1.312.852.0120
E-mail Address:	jbock@accuvant.com
Project Involvement:	Project oversight and quality assurance.

Name:	Phil Brass
Title:	Practice Manager –Software Security Group – Accuvant LABS
Phone:	Mobile: +1.678.296.5568
E-mail Address:	pbrass@accuvant.com
Project Involvement:	Project oversight and quality assurance.

Name:	Bill Ward
Title:	Solutions Engineer
Phone:	Mobile: +1.312.593.2455
E-mail Address:	bward@accuvant.com
Project Involvement:	Pre-sales technical contact

Name:	Leslie Ross
Title:	Account Manager
Phone:	Mobile: +1.408.398.6534
E-mail Address:	lross@accuvant.com
Project Involvement:	Sales contact

Appendix B – Methodology Overview

This section provides a brief overview of the methodologies Accuvant LABS uses for each phase of the assessment.

Application Profiling

Application profiling is the process of reviewing the available documentation and performing active analysis to determine how the application functions. User facing and developer documentation is reviewed, along with mapping the major application use cases. The application technology stack information is obtained either via communication with the development team or via technical analysis from interaction with the application. The user group and role matrix is considered, along with other critical aspects of the security model. Finally, key properties of the application's security controls are documented.

Threat Analysis

Accuvant LABS works with the application owners to identify critical data and functionality in the application that could be a target for attackers. This data is identified along with notes regarding its use and location within the application environment. The assessment team documents threat scenarios based on information from the development team as well as their own experience with application security.

Dynamic Testing

Dynamic testing procedures assess a running instance of the application for vulnerabilities. This is primarily a manual effort, with some support from assessment tools. The manual tests included review of the following:

- Direct interaction with the application (anonymous)
- Direct interaction with the application (privileged access with credentials)
- Review of system configurations
- Third party API interfaces and services

Further testing components may include:

- Application Environment Testing – Network vulnerability assessment of server and network platforms identify any issues in the application's supporting infrastructure.
- Unauthenticated Testing – Conduct tests with application assessment tools without authenticating to simulate an unauthenticated attacker. This includes a manual walkthrough of the application target with the assessment tool to configure the test run.
- Authenticated Testing – Conduct tests with user accounts included in the scope of the assessment. These include manual walkthroughs of the application target with the assessment tool to configure the test runs.

Below is a listing of some of the high-level categories that may be included in application testing. This is not a comprehensive list of all the areas to be tested in the assessment.

- Abuse of Functionality
- API Abuse
- Application Automation
- Audit and Logging Controls
- Authentication Deficiencies
- Authorization Deficiencies
- Buffer Overflows
- Command Execution
- Data Privacy and Integrity Controls
- Data Validation and Sanitization
- Encryption Implementations and Key Management
- Endpoint and Client Security Controls
- File and I/O Handling
- Injection Based Flaws

- Race Conditions
- Secure Channel Enforcement
- Service Configuration and Security
- Session Management
- UI and Content Security

The last step in this phase is to validate and potentially exploit the vulnerabilities identified through a proof of concept. This activity helps demonstrate the impact a particular issue can have on the environment, allows for further system penetration, and aids in identifying false positives.

Appendix C – Remote Testing Results

This section provides a summary of the results of the remote network-based testing operation. A more comprehensive spreadsheet with details about these findings has been distributed with this report.

Severity	IP Address	Affected Port	Host/DNS Name	Finding Title
Low	54.225.135.135	443	(cookiemonster.euclidelements.com)	HTTP Cookie 'secure' Property Transport Mismatch
Low	54.225.220.253	443	(cm-dev.euclidelements.com)	HTTP Cookie 'secure' Property Transport Mismatch
Low	54.243.55.69	443	(cm-beta.euclidelements.com)	HTTP Cookie 'secure' Property Transport Mismatch
Low	54.91.59.185	22	(ec2-54-91-59-185.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.83.130.84	22	(ec2-54-83-130-84.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.81.63.167	22	(ec2-54-81-63-167.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.81.199.117	22	(ec2-54-81-199-117.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.81.174.214	22	(ec2-54-81-174-214.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.80.22.20	22	(ec2-54-80-22-20.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.237.88.136	22	(ec2-54-237-88-136.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.237.115.49	22	(ec2-54-237-115-49.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.227.142.193	22	(ec2-54-227-142-193.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.226.109.219	22	(ec2-54-226-109-219.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.225.62.178	22	(ec2-54-225-62-178.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.225.193.242	22	(ec2-54-225-193-242.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.224.229.219	22	(ec2-54-224-229-219.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.224.168.104	22	(ec2-54-224-168-104.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.221.79.78	22	(ec2-54-221-79-78.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.221.31.121	22	(ec2-54-221-31-121.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.211.64.119	22	(ec2-54-211-64-119.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.211.40.21	22	(ec2-54-211-40-21.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.205.132.184	22	(ec2-54-205-132-184.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled

Low	54.198.75.126	22	(ec2-54-198-75-126.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.196.55.163	22	(ec2-54-196-55-163.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.165.64.7	22	(ec2-54-165-64-7.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.160.12.14	22	(ec2-54-160-12-14.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	50.19.4.195	22	(ec2-50-19-4-195.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	50.17.111.241	22	(ec2-50-17-111-241.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	23.23.125.65	22	(ec2-23-23-125-65.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	23.22.245.228	22	(ec2-23-22-245-228.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	23.22.217.225	22	(ec2-23-22-217-225.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	23.22.209.93	22	(ec2-23-22-209-93.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	184.73.19.212	22	(ec2-184-73-19-212.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.83.7.202	443	(express.euclidelements.com)	SSL Cipher Block Chaining Cipher Suites Supported
Low	23.23.212.195	443	(ex-dev.euclidelements.com)	SSL Cipher Block Chaining Cipher Suites Supported
Low	54.235.120.234	443	(ex-beta.euclidelements.com)	SSL Cipher Block Chaining Cipher Suites Supported
Low	54.243.163.157	443	(ec2-54-243-163-157.compute-1.amazonaws.com)	SSL Cipher Block Chaining Cipher Suites Supported
Low	54.243.119.204	443	(ec2-54-243-119-204.compute-1.amazonaws.com)	SSL Cipher Block Chaining Cipher Suites Supported
Low	54.235.185.61	443	(ec2-54-235-185-61.compute-1.amazonaws.com)	SSL Cipher Block Chaining Cipher Suites Supported
Low	54.197.241.124	443	(ec2-54-197-241-124.compute-1.amazonaws.com)	SSL Cipher Block Chaining Cipher Suites Supported
Low	50.19.242.125	443	(ec2-50-19-242-125.compute-1.amazonaws.com)	SSL Cipher Block Chaining Cipher Suites Supported
Low	174.129.215.176	443	(ec2-174-129-215-176.compute-1.amazonaws.com)	SSL Cipher Block Chaining Cipher Suites Supported
Low	54.225.135.135	443	(cookiemonster.euclidelements.com)	SSL Cipher Block Chaining Cipher Suites Supported
Low	54.225.220.253	443	(cm-dev.euclidelements.com)	SSL Cipher Block Chaining Cipher Suites Supported
Low	54.243.55.69	443	(cm-beta.euclidelements.com)	SSL Cipher Block Chaining Cipher Suites Supported
Low	54.83.7.202	443	(express.euclidelements.com)	SSL RC4 Cipher Suites Supported
Low	23.23.212.195	443	(ex-dev.euclidelements.com)	SSL RC4 Cipher Suites Supported
Low	54.235.120.234	443	(ex-beta.euclidelements.com)	SSL RC4 Cipher Suites Supported
Low	54.243.163.157	443	(ec2-54-243-163-157.compute-1.amazonaws.com)	SSL RC4 Cipher Suites Supported
Low	54.243.119.204	443	(ec2-54-243-119-204.compute-1.amazonaws.com)	SSL RC4 Cipher Suites Supported
Low	54.235.185.61	443	(ec2-54-235-185-61.compute-1.amazonaws.com)	SSL RC4 Cipher Suites Supported

Low	54.197.241.124	443	(ec2-54-197-241-124.compute-1.amazonaws.com)	SSL RC4 Cipher Suites Supported
Low	50.19.242.125	443	(ec2-50-19-242-125.compute-1.amazonaws.com)	SSL RC4 Cipher Suites Supported
Low	174.129.215.176	443	(ec2-174-129-215-176.compute-1.amazonaws.com)	SSL RC4 Cipher Suites Supported
Low	54.225.135.135	443	(cookiemonster.euclidelements.com)	SSL RC4 Cipher Suites Supported
Low	54.225.220.253	443	(cm-dev.euclidelements.com)	SSL RC4 Cipher Suites Supported
Low	54.243.55.69	443	(cm-beta.euclidelements.com)	SSL RC4 Cipher Suites Supported
Low	54.225.135.135	443	(cookiemonster.euclidelements.com)	HTTP Cookie 'secure' Property Transport Mismatch
Low	54.225.220.253	443	(cm-dev.euclidelements.com)	HTTP Cookie 'secure' Property Transport Mismatch
Low	54.243.55.69	443	(cm-beta.euclidelements.com)	HTTP Cookie 'secure' Property Transport Mismatch
Low	54.91.59.185	22	(ec2-54-91-59-185.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.83.130.84	22	(ec2-54-83-130-84.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.81.63.167	22	(ec2-54-81-63-167.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.81.199.117	22	(ec2-54-81-199-117.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.81.174.214	22	(ec2-54-81-174-214.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.80.22.20	22	(ec2-54-80-22-20.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.237.88.136	22	(ec2-54-237-88-136.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled
Low	54.237.115.49	22	(ec2-54-237-115-49.compute-1.amazonaws.com)	SSH Server CBC Mode Ciphers Enabled



About Accuvant

Accuvant is the only research-driven information security partner delivering alignment between IT security and business objectives, clarity to complex security challenges, and confidence in complex security decisions.

Based on our clients' unique requirements, Accuvant assesses, architects and implements the policies, procedures and technologies that most efficiently and effectively protect valuable data assets.

Since 2002, more than 4,500 organizations, including half of the Fortune 100 and 800 federal, state and local entities, have trusted Accuvant with their security challenges. Headquartered in Denver, Accuvant has offices across the United States and Canada. For more information, please visit www.accuvant.com, follow us on Twitter: @Accuvant, or keep in touch via Facebook: <http://tiny.cc/facebook553>.