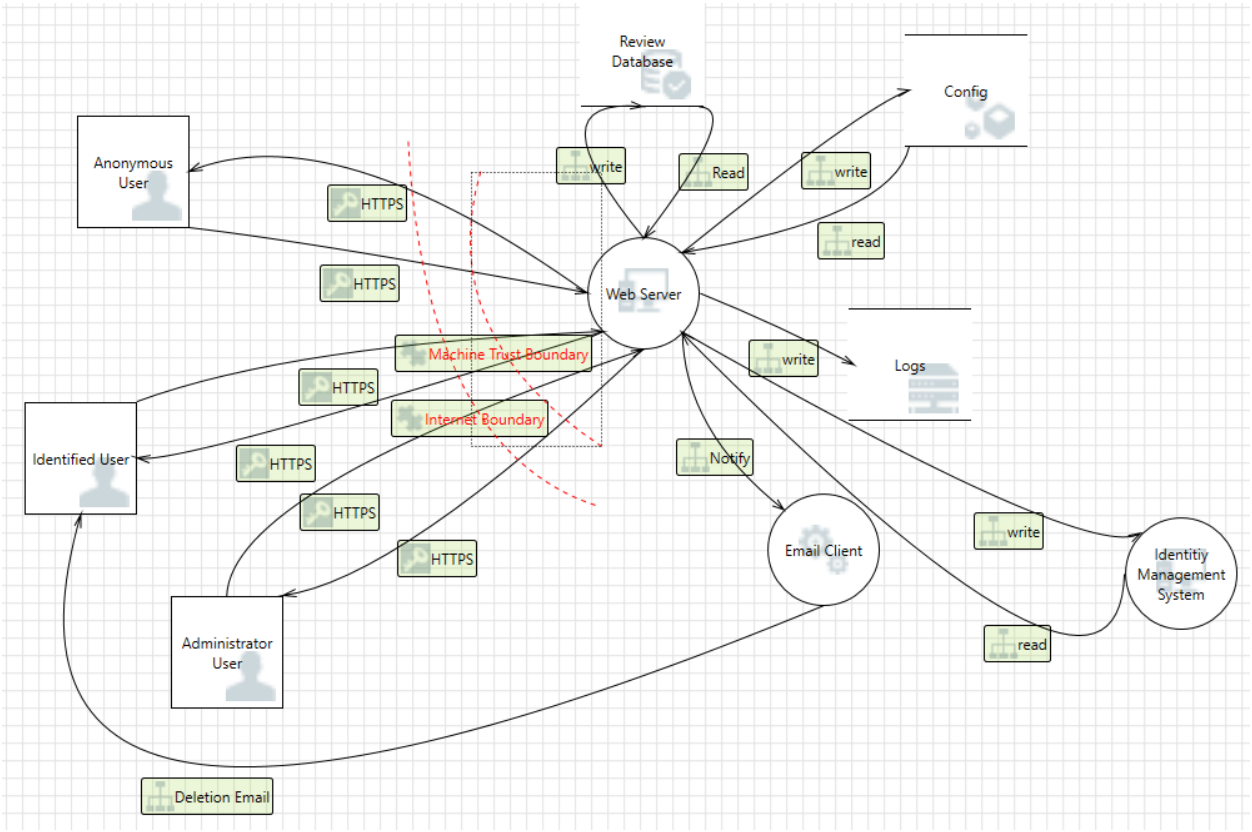
James Van Gilder: 9081186117

Model:



45 threats:

Title	Location	Description	Mitigation
Potential Data Repudiation by Web Server	HTTPS transfer from Identified User to Web Server	Web Server claims that it did not receive data from a source outside the trust boundary. Consider using logging or auditing to record the source, time, and summary of the received data.	MITIGATED BY LOGGING **ALREADY DONE**
Potential Process Crash or Stop for Web Server	HTTPS transfer from Identified User to Web Server	Web Server crashes, halts, stops or runs slowly; in all cases violating an availability metric.	Limit user requests or size of requests to avoid overloaded system

Data Flow HTTPS Is Potentially Interrupted	HTTPS transfer from Identified User to Web Server	An external agent interrupts data flowing across a trust boundary in either direction.	Disallow any data flow interruption via continued verification
Elevation Using Impersonation	HTTPS transfer from Identified User to Web Server	Web Server may be able to impersonate the context of Identified User in order to gain additional privilege.	Utilize server to user authentication such as nonce or other private cookies
Web Server May be Subject to Elevation of Privilege Using Remote Code Execution	HTTPS transfer from Identified User to Web Server	Identified User may be able to remotely execute code for Web Server.	Sanitize all input from all users prior to execution
Elevation by Changing the Execution Flow in Web Server	HTTPS transfer from Identified User to Web Server	An attacker may pass data into Web Server in order to change the flow of program execution within Web Server to the attacker's choosing.	Sanitize all input from all users prior to execution
Spoofing of the Identified User External Destination Entity	HTTPS transfer from Web Server to Identified User	Identified User may be spoofed by an attacker and this may lead to data being sent to the attacker's target instead of Identified User. Consider using a standard authentication mechanism to identify the external entity.	Utilize server to user authentication such as nonce or other private cookies
External Entity Identified User Potentially Denies Receiving Data	HTTPS transfer from Web Server to Identified User	Identified User claims that it did not receive data from a process on the other side of the trust boundary. Consider using logging or auditing to record the source, time, and summary of the received data.	MITIGATED BY LOGGING **ALREADY DONE**

Data Flow HTTPS Is Potentially Interrupted	HTTPS transfer from Web Server to Identified User	An external agent interrupts data flowing across a trust boundary in either direction.	Disallow any data flow interruption via continued verification
Potential Data Repudiation by Web Server	HTTPS transfer from Administrator User to Web Server	Web Server claims that it did not receive data from a source outside the trust boundary. Consider using logging or auditing to record the source, time, and summary of the received data.	MITIGATED BY LOGGING **ALREADY DONE**
Potential Process Crash or Stop for Web Server	HTTPS transfer from Administrator User to Web Server	Web Server crashes, halts, stops or runs slowly; in all cases violating an availability metric.	Limit user requests or size of requests to avoid overloaded system
Data Flow HTTPS Is Potentially Interrupted	HTTPS transfer from Administrator User to Web Server	An external agent interrupts data flowing across a trust boundary in either direction.	Disallow any data flow interruption via continued verification
Elevation Using Impersonation	HTTPS transfer from Administrator User to Web Server	Web Server may be able to impersonate the context of Administrator User in order to gain additional privilege.	Utilize server to user authentication such as nonce or other private cookies
Web Server May be Subject to Elevation of Privilege Using Remote Code Execution	HTTPS transfer from Administrator User to Web Server	Administrator User may be able to remotely execute code for Web Server.	Sanitize all input from all users prior to execution
Elevation by Changing the Execution Flow in Web Server	HTTPS transfer from Administrator User to Web Server	An attacker may pass data into Web Server in order to change the flow of program execution within Web Server to the attacker's choosing.	Sanitize all input from all users prior to execution
Spoofing of the	HTTPS transfer from	Anonymous User	Utilize server to user

Anonymous User External Destination Entity	Anonymous User to Web Server	may be spoofed by an attacker and this may lead to data being sent to the attacker's target instead of Anonymous User. Consider using a standard authentication mechanism to identify the external entity.	authentication such as nonce or other private cookies
External Entity Anonymous User Potentially Denies Receiving Data	HTTPS transfer from Anonymous User to Web Server	Anonymous User claims that it did not receive data from a process on the other side of the trust boundary. Consider using logging or auditing to record the source, time, and summary of the received data.	MITIGATED BY LOGGING **ALREADY DONE**
Data Flow HTTPS Is Potentially Interrupted	HTTPS transfer from Anonymous User to Web Server	An external agent interrupts data flowing across a trust boundary in either direction.	Disallow any data flow interruption via continued verification
Potential Data Repudiation by Web Server	HTTPS transfer from Web Server to Anonymous User	Web Server claims that it did not receive data from a source outside the trust boundary. Consider using logging or auditing to record the source, time, and summary of the received data.	MITIGATED BY LOGGING **ALREADY DONE**
Potential Process Crash or Stop for Web Server	HTTPS transfer from Web Server to Anonymous User	Web Server crashes, halts, stops or runs slowly; in all cases violating an availability metric.	Sanitize all input from all users prior to execution
Data Flow HTTPS Is Potentially	HTTPS transfer from Web Server to	An external agent interrupts data	Disallow any data flow interruption via

Interrupted	Anonymous User	flowing across a trust boundary in either direction.	continued verification
Elevation Using Impersonation	HTTPS transfer from Web Server to Anonymous User	Web Server may be able to impersonate the context of Anonymous User in order to gain additional privilege.	Utilize server to user authentication such as nonce or other private cookies
Web Server May be Subject to Elevation of Privilege Using Remote Code Execution	HTTPS transfer from Web Server to Anonymous User	Anonymous User may be able to remotely execute code for Web Server.	Sanitize all input from all users prior to execution
Elevation by Changing the Execution Flow in Web Server	HTTPS transfer from Web Server to Anonymous User	An attacker may pass data into Web Server in order to change the flow of program execution within Web Server to the attacker's choosing.	Sanitize all input from all users prior to execution
Spoofing of the Administrator User External Destination Entity	HTTPS transfer from Web Server to Administrator User	Administrator User may be spoofed by an attacker and this may lead to data being sent to the attacker's target instead of Administrator User. Consider using a standard authentication mechanism to identify the external entity.	Utilize server to user authentication such as nonce or other private cookies
External Entity Administrator User Potentially Denies Receiving Data	HTTPS transfer from Web Server to Administrator User	Administrator User claims that it did not receive data from a process on the other side of the trust boundary. Consider using logging or auditing to record the source, time, and summary of the received data.	MITIGATED BY LOGGING **ALREADY DONE**

Data Flow HTTPS Is Potentially Interrupted	HTTPS transfer from Web Server to Administrator User	An external agent interrupts data flowing across a trust boundary in either direction.	
Spoofing of Source Data Store Review Database	Data transfer from Web Server to Review Database	Review Database may be spoofed by an attacker and this may lead to incorrect data delivered to Web Server. Consider using a standard authentication mechanism to identify the source data store.	Verify at Database that all data is coming from only the Web Server using cookies
Weak Access Control for a Resource	Data transfer from Web Server to Review Database	Improper data protection of Review Database can allow an attacker to read information not intended for disclosure. Review authorization settings.	Encrypt all data stored
Spoofing of Destination Data Store Review Database	Data transfer from Review Database to Web Server	Review Database may be spoofed by an attacker and this may lead to data being written to the attacker's target instead of Review Database. Consider using a standard authentication mechanism to identify the destination data store.	Verify that location for sending data is correct via cookies
Potential SQL Injection Vulnerability for Review Database	Data transfer from Review Database to Web Server	SQL injection is an attack in which malicious code is inserted into strings that are later passed to an instance of SQL Server for parsing and execution. Any	Sanitize all user input, use prepared statements

		procedure that constructs SQL statements should be reviewed for injection vulnerabilities because SQL Server will execute all syntactically valid queries that it receives. Even parameterized data can be manipulated by a skilled and determined attacker.	
Potential Excessive Resource Consumption for Web Server or Review Database	Data transfer from Review Database to Web Server	Does Web Server or Review Database take explicit steps to control resource consumption? Resource consumption attacks can be hard to deal with, and there are times that it makes sense to let the OS do the job. Be careful that your resource requests don't deadlock, and that they do timeout.	Time out requests that take too long
Spoofing of Destination Data Store Logs	Data transfer from Web Server to Logs	Logs may be spoofed by an attacker and this may lead to data being written to the attacker's target instead of Logs. Consider using a standard authentication mechanism to identify the destination data store.	Verify that all logs being written are from the web server via cookies
Risks from Logging	Data transfer from Web Server to Logs	Log readers can come under attack via log files. Consider ways to canonicalize data in all logs.	Verify that only sanitized and approved log info from the web server can be written to the

		Implement a single reader for the logs, if possible, in order to reduce attack surface area. Be sure to understand and document log file elements which come from untrusted sources.	log file.
Lower Trusted Subject Updates Logs	Data transfer from Web Server to Logs	If you have trust levels, is anyone other outside of the highest trust level allowed to log? Letting everyone write to your logs can lead to repudiation problems. Only allow trusted code to log.	Nothing is allowed to log except for the verified web server
Data Logs from an Unknown Source	Data transfer from Web Server to Logs	Do you accept logs from unknown or weakly authenticated users or systems? Identify and authenticate the source of the logs before accepting them.	The only system that can write to log is the web server which is strongly authenticated
Insufficient Auditing	Data transfer from Web Server to Logs	Does the log capture enough data to understand what happened in the past? Do your logs capture enough data to understand an incident after the fact? Is such capture lightweight enough to be left on all the time? Do you have enough data to deal with repudiation claims? Make sure you log sufficient and appropriate data to handle a repudiation	Log captures sufficient data

		claims. You might want to talk to an audit expert as well as a privacy expert about your choice of data.	
Potential Weak Protections for Audit Data	Data transfer from Web Server to Logs	Consider what happens when the audit mechanism comes under attack, including attempts to destroy the logs, or attack log analysis programs. Ensure access to the log is through a reference monitor, which controls read and write separately. Document what filters, if any, readers can rely on, or writers should expect	Access to the logs are only given to valid, authenticated system administrators
Potential Excessive Resource Consumption for Web Server or Logs	Data transfer from Web Server to Logs	Does Web Server or Logs take explicit steps to control resource consumption? Resource consumption attacks can be hard to deal with, and there are times that it makes sense to let the OS do the job. Be careful that your resource requests don't deadlock, and that they do timeout.	Timeout requests that are taking too long
Weak Authentication Scheme	Data transfer from Web Server to Email Client	Custom authentication schemes are susceptible to common weaknesses such as weak credential change management,	Use standard, verified authentication scheme

		credential equivalence, easily guessable credentials, null credentials, downgrade authentication or a weak credential change management system. Consider the impact and potential mitigations for your custom authentication scheme.	
Elevation Using Impersonation	Data transfer from Web Server to Email Client	Email Client may be able to impersonate the context of Web Server in order to gain additional privilege.	Verify identity of the email client via cookies prior to further communication
Spoofing of Source Data Store Config	Data transfer from Config to Web Server	Config may be spoofed by an attacker and this may lead to incorrect data delivered to Web Server. Consider using a standard authentication mechanism to identify the source data store.	Implement standard authentication mechanism
Weak Access Control for a Resource	Data transfer from Config to Web Server	Improper data protection of Config can allow an attacker to read information not intended for disclosure. Review authorization settings.	Encrypt config and verify that only the verified administrator can access
Spoofing of Destination Data Store Config	Data transfer from Web Server to Config	Config may be spoofed by an attacker and this may lead to data being written to the attacker's target instead of Config.	Implement standard authentication mechanism

		Consider using a standard authentication mechanism to identify the destination data store.	
Elevation Using Impersonation	Data transfer from Web Server to Identity Management System	Identity Management System may be able to impersonate the context of Web Server in order to gain additional privilege.	Verify identity of the Identity Management System via cookies prior to further communication
Elevation Using Impersonation	Data transfer from Identity Management System to Web Server	Web Server may be able to impersonate the context of Identity Management System in order to gain additional privilege.	Verify identity of the Identity Management and Web Server System via cookies prior to further communication

Weakest Point in System:

The weakest point in my system is definitely my entity verification, it is very difficult to properly depict verification in this program, so I was unable to get my verification issues solved.

Should I Improve:

Verification and logging could use some improvement, ensuring via cookies like nonces would greatly help my system avoid attacks.

Was this useful:

I thought that the output from this tool was rather useful. It reminded me that sometimes it isn't even the simple user that is spoofed but rather the whole server, allowing access to things like logs and config files via those who can spoof the server. I also thought that it was useful to keep in mind just how many steps in the process can be attack points.