Appendix B

Humongous Insurance Price Quote Website

The Humongous Insurance Price Quote Website threat model presents some of the threats and issues that might be found in a Web application. This website is purposely limited in functionality to present threat modeling concepts without delving into the technology and product-specific issues that might be present in a modern Web application's threat model.

Table B-1 contains high-level information about the threat model being developed for the Price Quote Website application. This basic information includes the type of product and its location, the owner of the threat model and the team members, and any available milestone information.

Table B-1 Threat Model Information

Product	Price Quote Website
Milestone	Version 1.0
Owner	Darin Lockert
Participants	Don Funk, Jeffrey L. Ford, Luca Dellamore
Reviewer	Janice Galvin
Location	Humongous Insurance\Humongous Insurance Price Quote Website
Description	Humongous Insurance created an insurance price quote website to serve the needs of the company's growing online user base. This initial version of the website application has limited functionality. The application allows a user to create a profile (which includes a login and associated user data) that an insurance agent can review and use to respond with an insurance quote.

Use Scenarios

Table B-2 lists the known use scenarios for the application. This table provides information about the expected use of the application. Using or deploying the application in a way that violates a use scenario can impact the security of the application.

Table B-2 Use Scenarios

ID	Description
1	The Price Quote Website application will be installed on a Web server that has been secured to current industry guidelines. Current security patches for the Web server must be maintained.
2	The Price Quote Website will be installed on a database server that has been secured to current industry guidelines. Current security patches for the database server must be maintained.
3	The database server should be protected from direct access from the Internet by a firewall.
4	The Web server should be protected from direct access (except for the HTTP and HTTPS ports) from the Internet by a firewall.
5	Communication between the Web server and the database server should be conducted over a private network.
6	The Price Quote Website application should be deployed over HTTPS, except for the Welcome page, which might be accessible via HTTP.

External Dependencies

Table B-3 lists the external dependencies the application has on other components or products that can impact security. These dependencies are assumptions made about the usage or behavior of those other components or products. Inconsistencies can lead to security weaknesses in the Price Quote Website application.

Table B-3 External Dependencies

ID	Description
1	The Price Quote Website depends on the security of the Web server it is installed on. See Table B-2 for the corresponding use scenario.
2	The Price Quote Website depends on the security of the database server it is installed on. See Table B-2 for the corresponding use scenario.

Table B-3 External Dependencies

ID	Description
3	The Price Quote Website depends on the security of the network between the Web server and the database server. If this network is compromised, sensitive data could be viewed or direct attacks on the database server could be made.
4	The Price Quote Website depends on the session management of the Web server being secure. If the Web server's session management is not secure, an adversary might be able to hijack another user's session.
5	The Price Quote Website depends on an external Simple Mail Transfer Protocol (SMTP) server to deliver notifications of available quotes.

Implementation Assumptions

Table B-4 lists the application's implementation assumptions and describes the assumptions made about the internal workings of the component during the specification phase, but before implementation has started. These assumptions should not be violated. Typically, the threat modeling team will review them further once implementation is in place.

Table B-4 Implementation Assumptions

ID	Description
1	None. The Price Quote Website application is fully implemented.

External Security Notes

Table B-5 lists the external security notes—the threats or other information that an application user should be aware of to prevent possible vulnerabilities. These notes can include features that, if used incorrectly, could cause security problems for application users.

Table B-5 External Security Notes

ID	Description
1	The Price Quote Website application has no password quality enforcement. Users and insurance agents must choose strong passwords that are hard to guess or discover by brute force.
2	Insurance agent logins must be created directly through the database and cannot be created by the website. When creating insurance agent logins, the database administrator should ensure that communications to the database server are secure.

Internal Security Notes

Table B-6 lists the internal security notes, which contain security information that is relevant only to a person reading the threat model. These notes can be used to explain choices and design decisions that impact the product's security but were made due to overriding business needs.

Table B-6 Internal Security Notes

ID	Description
1	Because the Price Quote Website application does not use integrated authentication of any kind, end-to-end authentication and identity are not used. Supporting SQL authentication would delay the deployment of the website. In addition, the database in use supports connection pooling only when all connections use the same credentials. Thus, all queries to the database are done using one set of credentials (namely, the process identity of the Web server). Therefore, if an attack such as a SQL injection were to occur, the adversary would gain access to all tables in the database.
2	Humongous Insurance already has an SMTP server in its perimeter network (also known as the demilitarized zone or DMZ) that is used as a mail server for other Web applications. This system is controlled by another group, which is why the SMTP server is not considered in this threat model.

Trust Levels

Table B-7 lists the trust levels and describes privilege levels that are associated with entry points and assets.

Table B-7 Trust Levels

ID	Name	Description
1	Remote anonymous user	A user who has connected to the website but has not provided valid credentials yet.
2	Remote user with login credentials	A user who has created an account and has entered valid login credentials.
3	Insurance agent	Uses login credentials to view the Quote Review page.
4	Website administrator	An administrator that can configure the insurance quote website.
5	Database server administrator	An administrator that can access and modify the database and the information in it.
6	Web server process identity	Used to authenticate the Web server to the database when storing or retrieving information. All actions taken by the Web server process occur under this identity.
7	Database server process identity	The account that the database server process runs as, represented by its process token. The database process has all the access and privileges that correspond to this token.
8	HTTP user	A remote user that accesses a page via HTTP.
9	HTTPS user	A remote user that accesses a page via HTTPS.

Entry Points

Table B-8 lists the entry points and describes the interfaces through which external entities can interact with the component, either by direct interaction or by indirectly supplying it with data.

Table B-8 Entry Points

ID	Name	Description	Trust Level
1	Web server listening port (HTTPS)	The port (HTTPS) that the Web server listens on. Most of the site's Web pages are layered on this port.	(1) Remote anonymous user (2) Remote user with login credentials (3) Insurance agent (4) Website administrator (9) HTTPS user
1.1	Login page	Page for users to create a login and perform a login to the site to begin requesting or reviewing an insurance quote.	(1) Remote anonymous user (2) Remote user with login credentials (3) Insurance agent (9) HTTPS user
1.1.1	*CreateLogin* function	Creates a new user login. (Insurance agent logins must be created directly through the database stored procedures.)	(1) Remote anonymous user
1.1.2	*LoginToSite* function	Compares user-supplied credentials to those in the database and creates a new session if the credentials match.	(1) Remote anonymous user (2) Remote user with login credentials (3) Insurance agent
1.2	Data entry page	Page used to enter user's personal data into the database so that insurance agents can review it.	(2) Remote user with login credentials (9) HTTPS user
1.2.1	*RetrieveData* function	Allows the user to view his previously entered information along with the insurance quote, if available.	(2) Remote user with login credentials
1.2.2	*SubmitData* function	Submits user data to be reviewed by the insurance agent.	(2) Remote user with login credentials
1.3	Insurance agent Quote Review page	Page used by insurance agents to review a user's request for a quote. Agents also use this page to enter insurance quote information.	(3) Insurance agent (9) HTTPS user

Table B-8 Entry Points

ID	Name	Description	Trust Level
1.3.1	*RetrieveData* function	Retrieves user data so that the insurance agent can generate an insurance quote.	(3) Insurance agent
1.3.2	*SubmitData* function	Submits an insurance quote for the user to review.	(3) Insurance agent
1.3.3	*ListRequests* function	Lists quote requests ready for review.	(3) Insurance agent
2	Database listening port	Enables the database to be used remotely.	(1) Remote anonymous user (5) Database server administrator (6) Web server process identity (7) Database server process identity
2.1	Database stored procedures	Store and retrieve quote-related information in the database.	(1) Remote anonymous user (2) Remote user with login credentials (3) Insurance agent (5) Database server administrator (6) Web server process identity
2.1.1	*CreateLogin* procedure	Creates a website login for a user or an insurance agent.	(1) Remote anonymous user (5) Database server administrator (6) Web server process identity
2.1.2	*RemoveLogin* procedure	Removes a login, including any user data, if the login is a user rather than an insurance agent.	(5) Database server administrator (6) Web server process identity

Table B-8 Entry Points

ID	Name	Description	Trust Level
2.1.3	*StoreUserData* procedure	Used to store user data from the data entry page of the website.	(2) Remote user with login credentials (5) Database server administrator (6) Web server process identity
2.1.4	*RetrieveUserData* procedure	Retrieves the user's data and insurance quote.	(2) Remote user with login credentials (3) Insurance agent (5) Database server administrator (6) Web server process identity
2.1.5	*StoreQuoteData* procedure	Used by the insurance agent Quote Review page to store an insurance quote for the user to view.	(3) Insurance agent (5) Database server administrator (6) Web server process identity
2.1.6	*RetrieveCredentials* procedure	Used to retrieve login credentials for a user or insurance agent. The website compares this information to the credentials that the user sends to the login page.	(5) Database server administrator (6) Web server process identity
3	Web pages from disk	The Web pages on disk are entry points. The Web server reads these files and uses the code in them to process requests.	(4) Website administrator (6) Web server process identity
4	Web server listening port (HTTP)	The HTTP port that the Web server listens on. Only the Welcome page is layered on this port.	(1) Remote anonymous user (8) HTTP user
4.1	Welcome page	A static HTML page that informs the user about the Price Quote Website and redirects him to the login page.	(1) Remote anonymous user (8) HTTP user
5	Connection to SMTP server	The Price Quote Website connects to an SMTP server to send notification e-mails.	(6) Web server process identity

Assets

Table B-9 lists the assets and describes the data or functionality that the component needs to protect. This table also lists the minimum access category (trust level) that should be allowed to access the resource.

Table B-9 Assets

ID	Name	Description	Trust Level
13	User and agent	Assets that relate to a user or insurance agent.	
13.1	User's login data	The user's credentials: username and password.	(2) Remote user with login credentials (5) Database server administrator
13.7	Insurance agent's login data	The insurance agent's credentials: username and password.	(3) Insurance agent (5) Database server administrator
13.2	User's personal data	The personal data that the user enters, such as contact information and assets.	(2) Remote user with login credentials (3) Insurance agent
14	System	Assets that relate to the underlying system.	
14.12	Availability of the site	If the Price Quote Website goes down, users cannot request or receive quotes.	(4) Website administrator (5) Database server administrator
15	Process	Assets that relate to the process running the website.	
15.4	Ability to execute code as the identity of the Web server	Web pages on the site execute code by using the security token of the Web server.	(4) Website administrator (6) Web server process identity
15.5	Ability to execute code as the identity of the database server	Database queries and procedures execute code using the security token of the database server.	(5) Database server administrator (7) Database server process identity
16	Application	Assets that relate to the Web application.	

Table B-9 Assets

ID	Name	Description	Trust Level
16.8	Login session	The Web session associated with a logged-in user or insurance agent.	(2) Remote user with login credentials (3) Insurance agent
16.3	Access to backend database	The ability to interact with the database that stores user data, insurance quotes, and login credentials.	(5) Database server administrator (6) Web server process identity (7) Database server process identity
16.6	Accuracy of the price quote	The insurance price quote must be accurate. An adversary tampering with this quote could cause loss of business.	(3) Insurance agent
16.9	Access to the insurance agent pages	Only insurance agents should be able to view the insurance agent pages.	(3) Insurance agent
16.10	E-mail notification of ready quote	The e-mail notification lets the user know when a quote is ready.	(2) Remote user with login credentials (3) Insurance agent
16.11	Notification of new quote request	Only quote requests in the notify list will be seen by insurance agents.	(2) Remote user with login credentials (3) Insurance agent
16.1	Audit data	Adversaries might try to attack the system without being logged or audited.	(4) Website administrator (5) Database server administrator (6) Web server process identity (7) Database server process identity

Data Flow Diagrams

Figure B-1 shows the context-level data flow diagram (DFD) for the Humongous Insurance Price Quote Website application. The system is represented as two processing nodes to denote a computer boundary. Users, agents, and servers external to the website are shown as external entities.

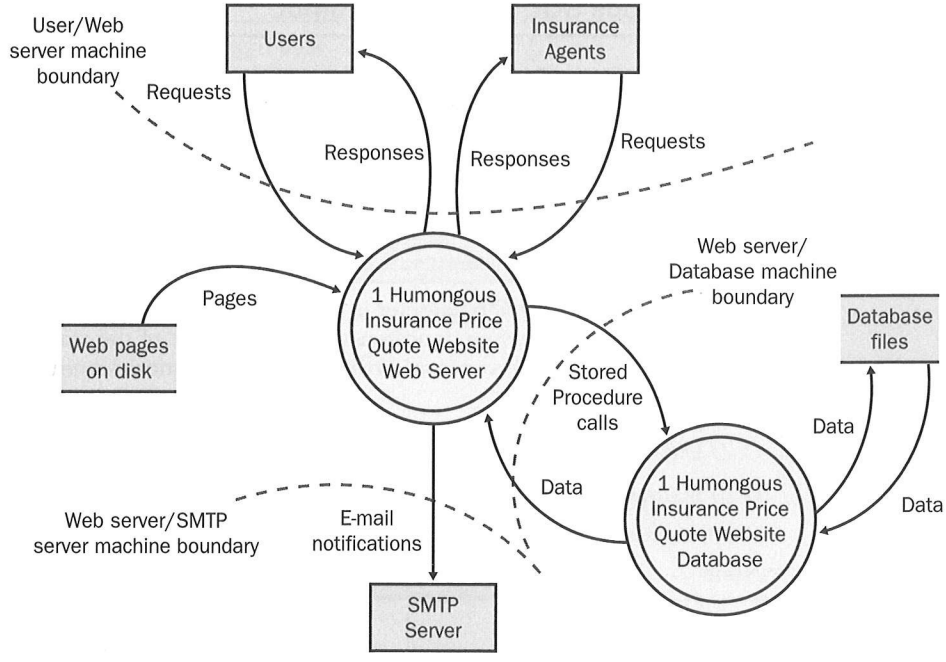


Figure B-1 Data flow diagram.

Threats

The threats to the application are listed in a series of tables—one for each threat. These threats do not imply vulnerabilities. Rather, they are actions that a malicious external entity might try to perform to exploit the system.

Table B-10 Threat: Malicious SQL Data in User Input

ID	1
Name	Adversary supplies malicious data in a request targeting the SQL command-parsing engine in an attempt to change execution
Description	An adversary might try to inject SQL commands into the application via data she supplies, such as her login name or personal information. If this data is not handled properly by the Price Quote Website, this could result in SQL injection. In addition, other malicious input could cause the system to become unstable or leak information.
STRIDE classification	■ Tampering ■ Elevation of privilege
Mitigated?	No

Table B-10 Threat: Malicious SQL Data in User Input

Known mitigation	None
Investigation notes	The database stored procedures were code reviewed for any use of string concatenation in freeform EXEC queries. The *RetrieveCredentials* stored procedure was the only procedure with this error. The procedure concatenates the *@username* parameter to a SELECT statement: `EXEC('SELECT Password FROM LoginTable WHERE Username = ' + @username)` Because the *@username* parameter is not validated, an attacker could supply a malicious string that, when concatenated with the rest of the statement and then reparsed by the SQL server, could result in arbitrary queries being run.
Entry points	(1.1) Login page (1.2) Data entry page (1.3) Insurance agent Quote Review page
Assets	(16.3) Access to backend database
Threat tree	None

Table B-11 Threat: Disclosure of Login Information

ID	2
Name	Adversary acquires the username and password or another user or agent
Description	If an adversary obtains the login credentials of another user or agent, he can do perform any task that user can.
STRIDE classification	■ Information disclosure ■ Elevation of privilege
Mitigated?	No
Known mitigation	*Related use scenarios:* (3) The database server should be protected from direct access from the Internet by a firewall. *Related external security notes:* (1) No password quality enforcement exists in the Price Quote Website. It is up to users and agents to choose strong passwords that are hard to guess or brute-force discover.
Investigation notes	None
Entry points	(1.1) Login page (2.1) Database stored procedures

Table B-11 Threat: Disclosure of Login Information

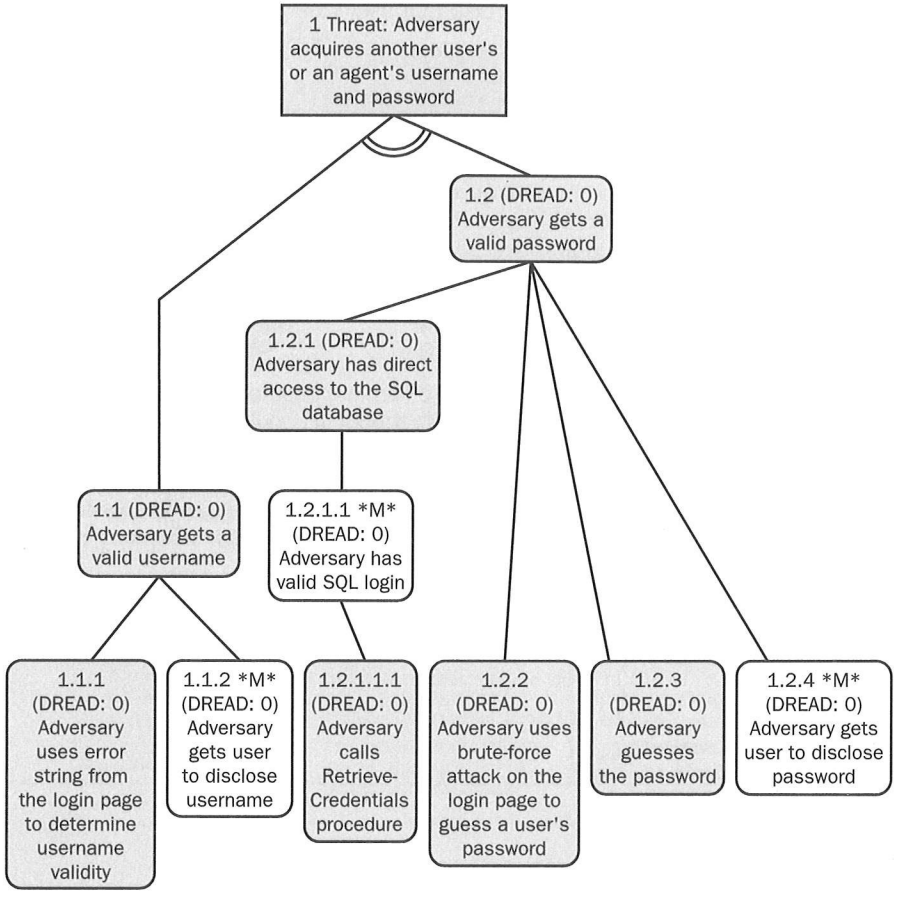
Assets	(13.1) User's login data (13.7) Insurance agent's login data
Threat tree	

Table B-12 Threat: Session ID Theft

ID	3
Name	Adversary acquires the session ID of another user or agent
Description	If an attacker acquires the session ID of a logged-in user or agent, she can perform any task that user can.
STRIDE classification	Elevation of privilege
Mitigated?	Yes
Known mitigation	The Price Quote Website depends on the cryptographic security of the Web server's session management. In addition, the session is established only after the user accesses the login page. Because this should be done over the Secure Socket Layer (SSL), an adversary should not be able to snoop the session identifier. (For more information, see the mitigating use scenario below.) *Related use scenarios:* (6) The Price Quote Website should be deployed over HTTPS, except for the Welcome page, which may accessible via HTTP. *Related external dependencies:* (4) The Price Quote Website depends on the session management of the Web server to be secure. If it is not, an adversary might be able to hijack another user's session.
Investigation notes	None
Entry points	(1) Web server listening port (HTTPS)
Assets	(16.8) Login session
Threat tree	None

Table B-13 Threat: User Data Disclosure

ID	4
Name	Adversary retrieves another user's personal data
Description	Disclosing another user's personal data raises privacy issues. Furthermore, Humongous Insurance would not be perceived as trustworthy.
STRIDE classification	■ Spoofing ■ Information disclosure
Mitigated?	No
Known mitigation	None

Table B-13 Threat: User Data Disclosure

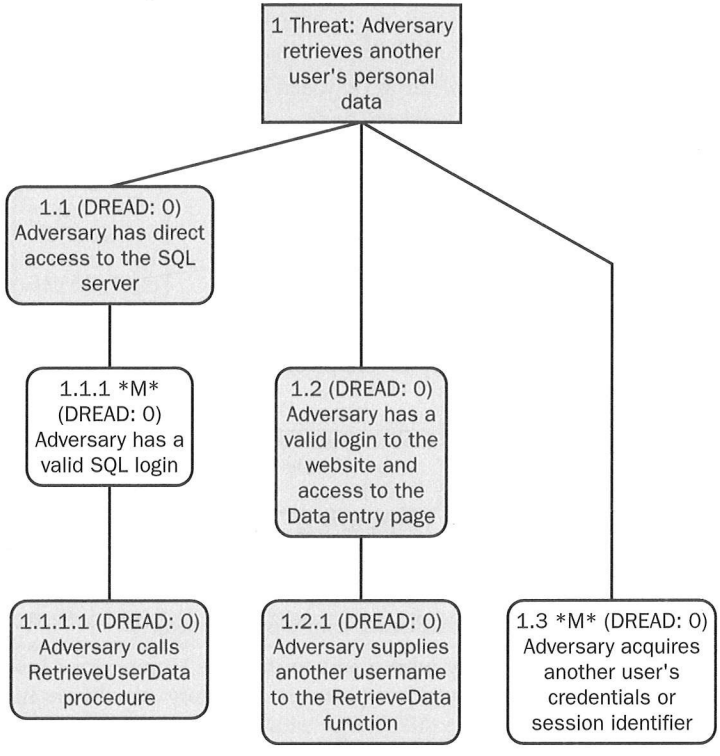
Investigation notes	None
Entry points	(1.2) Data entry page (2.1) Database stored procedures
Assets	(13.2) User's personal data
Threat tree	

Table B-14 Threat: Direct Access to the Database

ID	5
Name	Adversary accesses the backend database directly
Description	An adversary who can access the backend database directly might be able to tamper with or view the data stored in it, including account names and passwords.
STRIDE classification	■ Tampering ■ Repudiation ■ Information disclosure ■ Elevation of privilege

Table B-14 Threat: Direct Access to the Database

Mitigated?	Yes
Known mitigation	For this to be a vulnerability, the website would have to be deployed in an unsupported configuration or an adversary would have to be able to execute code on the Web server, which has network access to the database. *Related use scenarios:* (3) The database server should be protected from direct access from the Internet by a firewall. (5) Communication between the Web server and the database server should be over a private network.
Investigation notes	None
Entry points	(2) Database listening port
Assets	(13.1) User's login data (13.7) Insurance agent's login data (13.2) User's personal data (16.6) Accuracy of the price quote
Threat tree	None

Table B-15 Threat: Rate Quote Tampering

ID	6
Name	Adversary modifies a user's price quote
Description	An adversary who wants to damage Humongous Insurance's business might try to modify price quotes so that they are always higher than those of competitors.
STRIDE classification	Tampering
Mitigated?	Yes
Known mitigation	See corresponding threat tree in this table.
Investigation notes	None
Entry points	(1.3) Insurance agent Quote Review page (2.1) Database stored procedures
Assets	(16.6) Accuracy of the price quote

Table B-15 Threat: Rate Quote Tampering

Threat tree

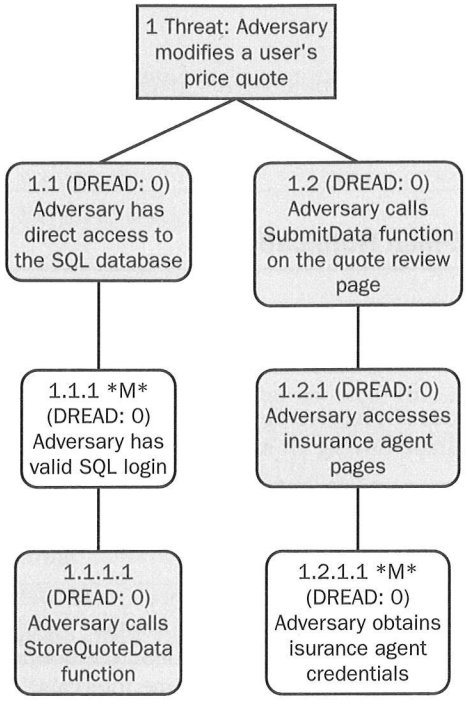


Table B-16 Threat: Unauthorized Use of Insurance Agent Web Pages

ID	7
Name	Adversary accesses insurance agent Web pages
Description	An adversary who can access the insurance agent Web pages might be able to view sensitive information.
STRIDE classification	Elevation of privilege
Mitigated?	Yes
Known mitigation	The insurance agent Web pages check server-side session state to determine whether the user associated with the session is an insurance agent. If the user is not an agent, the user is redirected to the login page.

Table B-16 Threat: Unauthorized Use of Insurance Agent Web Pages

Investigation notes	None
Entry points	(1.3) Insurance agent Quote Review page
Assets	(16.9) Access to the insurance agent Web pages
Threat tree	None

Table B-17 Threat: Blocking E-Mail Notifications

ID	8
Name	Adversary prevents a user from receiving e-mail notification of an available quote
Description	If the user does not receive the e-mail notification of the available quote, she might not pursue buying an insurance policy from Humongous Insurance.
STRIDE classification	Denial of service
Mitigated?	Yes
Known mitigation	The e-mail server is external to the Price Quote Website application. The security of that server is controlled by another group. If the website cannot contact the Web server, the site will queue the notification and try resending the e-mail at recurring, configurable time intervals. *Related external dependencies:* (5) The Price Quote Website depends on an external SMTP server to deliver notifications of ready quotes.
Investigation notes	None
Entry points	(5) Connection to SMTP server
Assets	(16.10) E-mail notification of available quote
Threat tree	None

Table B-18 Threat: Blocking New Quote Request Notifications

ID	9
Name	Adversary prevents insurance agents from receiving notification of new quote requests
Description	If the insurance agent does not receive notification that new requests are waiting, those requests will not be processed.
STRIDE classification	Denial of service
Mitigated?	Yes
Known mitigation	See corresponding investigation notes in this table.
Investigation notes	The list of quote requests is retrieved when an insurance agent logs in. No methods of blocking this list were found.
Entry points	(1.3) Insurance agent Quote Review page (2.1) Database stored procedures
Assets	(16.11) Notification of new quote request
Threat tree	None

Table B-19 Threat: User Data Tampering

ID	10
Name	Adversary modifies another user's personal data
Description	Modifying another user's data could alter that user's quoted price and would make Humongous Insurance look untrustworthy.
STRIDE classification	■ Spoofing ■ Tampering ■ Elevation of privilege
Mitigated?	No
Known mitigation	None
Investigation notes	See the threat tree in Table B-13—specifically, the threat of retrieving another user's personal data.
Entry points	(1.2) Data entry page (2.1) Database stored procedures
Assets	(13.2) User's personal data
Threat tree	None

Table B-20 Threat: Account Deletion

ID	11
Name	Adversary deletes a user or agent account
Description	This deletion of a user or agent account would cause a denial of service. The user or agent would no longer be able to work with the Price Quote Website.
STRIDE classification	■ Denial of service ■ Elevation of privilege
Mitigated?	Yes
Known mitigation	See the corresponding mitigation use scenarios below. *Related use scenarios:* (3) The database server should be protected from direct access from the Internet by a firewall. (5) Communication between the Web server and the database server should be over a private network.
Investigation notes	The website does not support deleting accounts directly. This must be done by connecting directly to the database and calling the *RemoveLogin* procedure.
Entry points	(2.1) Database stored procedures
Assets	(13.1) User's login data (13.7) Insurance agent's login data
Threat tree	None

Table B-21 Threat: Crashing the Website

ID	12
Name	Adversary crashes the Price Quote Website
Description	Crashing the website would cause a denial of service, preventing quotes from being requested or reviewed.
STRIDE classification	Denial of service
Mitigated?	Yes

Table B-21 Threat: Crashing the Website

Known mitigation	Stress testing the website is part of the test suite. In addition, the test suite has numerous invalid input tests to ensure the website can handle malformed data. *Related use scenarios:* (1) The Price Quote Website will be installed on a Web server that has been secured to current industry guidelines. Current security patches for the Web server must be maintained.
Investigation notes	None
Entry points	(1) Web server listening port (HTTPS) (2) Database listening port (4) Web server listening port (HTTP)
Assets	(14.12) Availability of the site
Threat tree	None

Table B-22 Threat: Accessing the Site Without Valid Credentials

ID	13
Name	Adversary without valid credentials accesses the site
Description	An adversary without any credentials might try to access a user's account or the insurance agent Web pages.
STRIDE classification	Elevation of privilege
Mitigated?	Yes
Known mitigation	All pages redirect to the login page if the server-side session state does not indicate a logged-in session. A session is marked as logged in only if the user-supplied password matches that in the database. *Related external dependencies:* (4) The Price Quote Website depends on the session management of the Web server to be secure. If it is not, an adversary might be able to hijack another user's session.
Investigation notes	None
Entry points	(1.1) Login page (1.2) Data entry page (1.3) Insurance agent Quote Review page
Assets	(13.2) User's personal data
Threat tree	None

Table B-23 Threat: Intercepting Available Quote Notification

ID	14
Name	Adversary intercepts e-mail notification of available quote
Description	The adversary would be able to view any sensitive data in the email.
STRIDE classification	Information disclosure
Mitigated?	Yes
Known mitigation	The information in the e-mail consists of the username and a message stating that the quote is ready. The e-mail message does not contain quote information or personal data. Thus, even if the message is intercepted, no sensitive data will be disclosed.
Investigation notes	None
Entry points	(5) Connection to SMTP server
Assets	(16.10) E-mail notification of available quote
Threat tree	None

Table B-24 Threat: Access Without Auditing

ID	15
Name	Adversary tries to access another user's data without being logged
Description	If no logging occurs, there is no way of knowing that the attack occurred and no evidence to indicate where the attack came from.
STRIDE classification	Repudiation
Mitigated?	No
Known mitigation	None
Investigation notes	Other than minimal built-in Web server logging, the website has no logging.
Entry points	(1.2) Data entry page
Assets	(16.1) Audit data
Threat tree	None

Vulnerabilities

The known vulnerabilities to the system are listed in a series of tables in this section—one table for each vulnerability. Each table includes the risk associated with not fixing the vulnerability so that threat modeling teams can choose mitigation strategies appropriately.

Table B-25 Vulnerability: SQL Injection

ID	1
Name	*RetrieveCredentials* SQL injection
Description	The *RetrieveCredentials* procedure concatenates the *@username* parameter to a SELECT statement: `EXEC('SELECT Password FROM LoginTable WHERE Username = ' + @username)` Because the *@username* parameter is not validated, an attacker could supply a malicious string that, when concatenated with the rest of the statement and then reparsed by the instance of SQL Server, could result in arbitrary queries being run. The *@username* parameter should be sanitized for malicious input, or a parameterized version should be used.
STRIDE classification	■ Tampering ■ Elevation of privilege
DREAD rating	10
Corresponding threat ID	1: Adversary uses special SQL characters or keywords in her input to attempt to execute code on the database server
Bug	223

Table B-26 Vulnerability: Username Discoverability

ID	2
Name	Confirmation of whether a username exists via login page error strings
Description	The login page returns two errors, "Bad username" and "Bad password." These errors allow an adversary to determine whether a username exists. Damage potential is low because this information alone does not grant the adversary access to the website application; the adversary still has to guess the password. In addition, exploitability is low because the adversary must guess the username as well.
STRIDE classification	■ Information disclosure ■ Elevation of privilege
DREAD rating	6.6
Corresponding threat ID	2: Adversary acquires the username and password or another user or agent
Bug	250

Table B-27 Vulnerability User Data Disclosure

ID	3
Name	User data disclosure in the *RetrieveData* function on the data entry page
Description	The *RetrieveData* function on the data entry page uses a parameter in the HTTP request (username) to determine which user's data to retrieve from the database. The username should instead be retrieved from server-side session state so that the client cannot arbitrarily supply the username.
STRIDE classification	■ Spoofing ■ Information disclosure
DREAD rating	9.2
Corresponding threat ID	4: Adversary retrieves another user's personal data
Bug	251

Table B-28 Vulnerability: Data Tampering

ID	4
Name	User tampers with data in the *SubmitData* function on the data entry page
Description	The *SubmitData* function on the data entry page uses a parameter in the HTTP request (username) to determine which user's data to store to the database. The username should instead be retrieved from server-side session state so that the client cannot arbitrarily supply the username.
STRIDE classification	■ Spoofing ■ Tampering ■ Elevation of privilege
DREAD rating	9.2
Corresponding Threat ID	10: Adversary modifies another user's personal data
Bug	259

Table B-29 Vulnerability: Nonexistent Logging and Auditing

ID	5
Name	The Price Quote Website has no logging or auditing
Description	Humongous Insurance needs to determine what functionality should be logged and audited. The company must also assess the legal ramifications of this.
STRIDE classification	Repudiation
DREAD rating	5.2
Corresponding threat ID	15: Adversary tries to access other user's data without being logged
Bug	260