Threat Modeling Report

Created on 11/7/2017 9:11:45 PM

Threat Model Name:

Owner:

Reviewer:

Contributors:

Description:

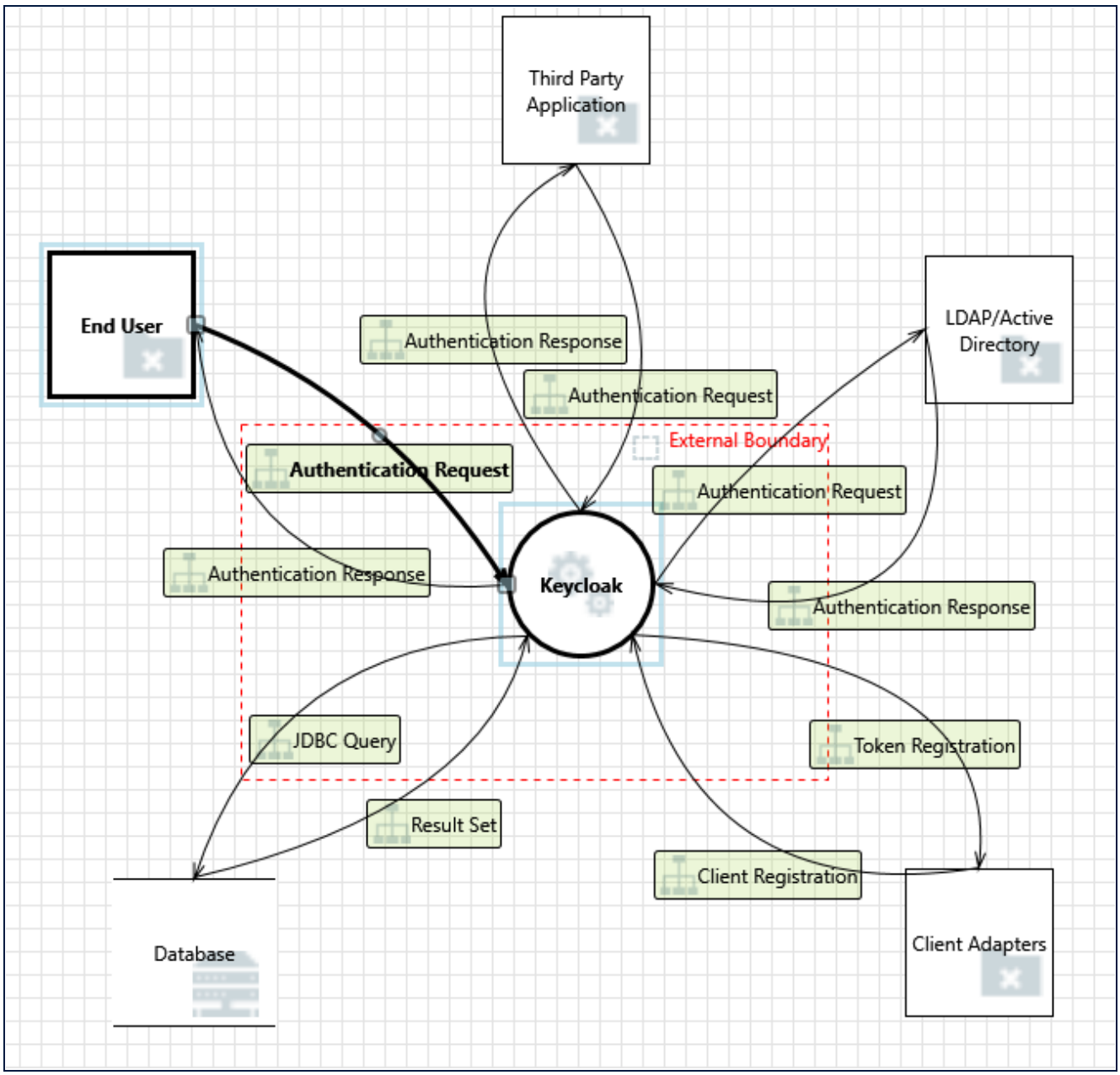
Assumptions:

External Dependencies:

Threat Model Summary:

Not Started	73
Not Applicable	0
Needs Investigation	0
Mitigation Implemented	0
Total	73
Total Migrated	0

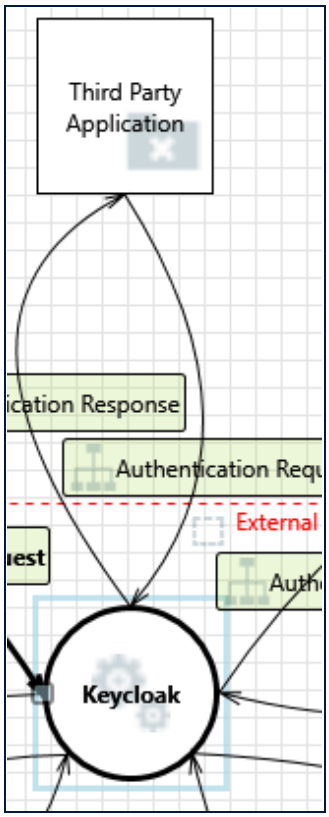
Diagram: Keycloak Threat Modeling



Keycloak Threat Modeling Diagram Summary:

Not Started	73
Not Applicable	0
Needs Investigation	0
Mitigation Implemented	0
Total	73
Total Migrated	0

Interaction: Authentication Request



1. Spoofing the Third Party Application External Entity [State: Not Started] [Priority: High]

Category: Spoofing

Description: Third Party Application may be spoofed by an attacker and this may lead to unauthorized access to Keycloak. Consider using a standard authentication mechanism to identify the external entity.

Justification: Use multi-factor authentication

2. Elevation Using Impersonation [State: Not Started] [Priority: High]

Category: Elevation Of Privilege

Description: Keycloak may be able to impersonate the context of Third Party Application in order to gain additional privilege.

Justification: Utilize principle of least privilege

3. Cross Site Request Forgery [State: Not Started] [Priority: High]

Category: Elevation Of Privilege

Description: Cross-site request forgery (CSRF or XSRF) is a type of attack in which an attacker forces a user's browser to make a forged request to a vulnerable site by exploiting an existing trust relationship between the browser and the vulnerable web site. In a simple scenario, a user is logged in to web site A using a cookie as a credential. The other browses to web site B. Web site B returns a page with a hidden form that posts to web site A. Since the browser will carry the user's cookie to web site A, web site B now can take any action on web site A, for example, adding an admin to an account. The attack can be used to exploit any requests that the browser automatically authenticates, e.g. by session cookie, integrated authentication, IP whitelisting, ... The attack can be carried out in many ways such as by luring the victim to a site under control of the attacker, getting the user to click a link in a phishing email, or hacking a reputable web site that the victim will visit. The issue can only be resolved on the server side by requiring that all authenticated state-changing requests include an additional piece of secret payload (canary or CSRF token) which is known only to the legitimate web site and the browser and which is protected in transit through SSL/TLS. See the Forgery Protection property on the flow stencil for a list of mitigations.

Justification: Require a state cookie be used and matched against a transmitted state parameter

4. Elevation by Changing the Execution Flow in Keycloak [State: Not Started] [Priority: High]

Category: Elevation Of Privilege

Description: An attacker may pass data into Keycloak in order to change the flow of program execution within Keycloak to the attacker's choosing.

Justification: Ulitize input validation and implement ACLs

5. Keycloak May be Subject to Elevation of Privilege Using Remote Code Execution [State: Not Started] [Priority: High]

Category: Elevation Of Privilege

Description: Third Party Application may be able to remotely execute code for Keycloak.

Justification: Vallidate all input

6. Data Flow Authentication Request Is Potentially Interrupted [State: Not Started] [Priority: High]

Category: Denial Of Service

Description: An external agent interrupts data flowing across a trust boundary in either direction.

Justification: Ability to continue processing in reduced capacity

7. Potential Process Crash or Stop for Keycloak [State: Not Started] [Priority: High]

Category: Denial Of Service
Description: Keycloak crashes, halts, stops or runs slowly; in all cases violating an availability metric.
Justification: Validate all input and ability to continue processing in a reduced capacity.

8. Data Flow Sniffing [State: Not Started] [Priority: High]

Category: Information Disclosure
Description: Data flowing across Authentication Request may be sniffed by an attacker. Depending on what type of data an attacker can read, it may be used to attack other parts of the system or simply be a disclosure of information leading to compliance violations. Consider encrypting the data flow.
Justification: Use encryption with authentication

9. Potential Data Repudiation by Keycloak [State: Not Started] [Priority: High]

Category: Repudiation
Description: Keycloak 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.
Justification: Logging and auditing of recieved data.

10. Potential Lack of Input Validation for Keycloak [State: Not Started] [Priority: High]

Category: Tampering
Description: Data flowing across Authentication Request may be tampered with by an attacker. This may lead to a denial of service attack against Keycloak or an elevation of privilege attack against Keycloak or an information disclosure by Keycloak. Failure to verify that input is as expected is a root cause of a very large number of exploitable issues. Consider all paths and the way they handle data. Verify that all input is verified for correctness using an approved list input validation approach.
Justification: Normalization before sanitization

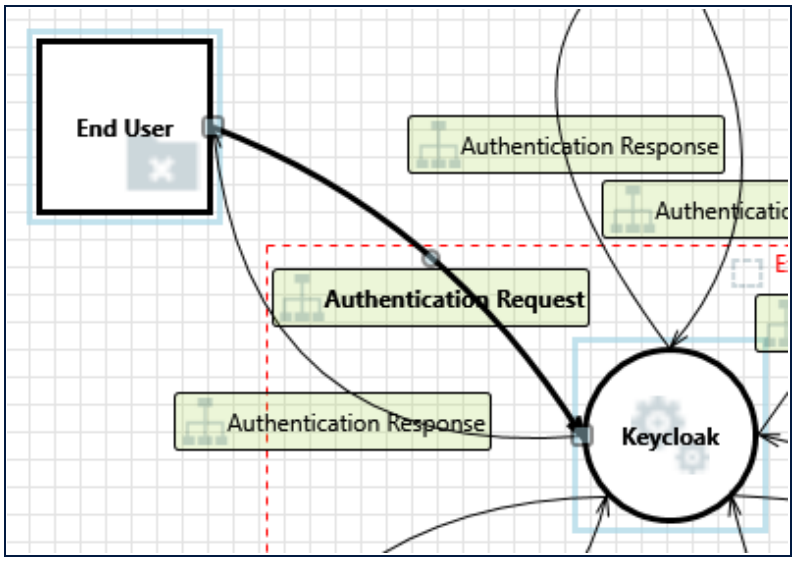
11. Spoofing the Keycloak Process [State: Not Started] [Priority: High]

Category: Spoofing
Description:

Keycloak may be spoofed by an attacker and this may lead to information disclosure by Third Party Application. Consider using a standard authentication mechanism to identify the destination process.

Justification: Use multi-factor authentication

Interaction: Authentication Request



12. Spoofing the End User External Entity [State: Not Started] [Priority: High]

Category: Spoofing

Description: End User may be spoofed by an attacker and this may lead to unauthorized access to Keycloak. Consider using a standard authentication mechanism to identify the external entity.

Justification: Utilize multi-factor authentication

13. Elevation Using Impersonation [State: Not Started] [Priority: High]

Category: Elevation Of Privilege

Description: Keycloak may be able to impersonate the context of End User in order to gain additional privilege.

Justification: Utilize principle of least privilege

14. Cross Site Request Forgery [State: Not Started] [Priority: High]

Category: Elevation Of Privilege

Description: Cross-site request forgery (CSRF or XSRF) is a type of attack in which an attacker forces a user's browser to make a forged request to a vulnerable site by exploiting an existing trust relationship between the browser and the vulnerable web site. In a simple scenario, a user is logged in to web site A using a cookie as a credential. The other browses to web site B. Web site B returns a page with a hidden form that posts to web site A. Since the browser will carry the user's cookie to web site A, web site B now can take any action on web site A, for example, adding an admin to an account. The attack can be used to exploit any requests that the browser automatically authenticates, e.g. by session cookie, integrated authentication, IP whitelisting, ... The attack can be carried out in many ways such as by luring the victim to a site under control of the attacker, getting the user to click a link in a phishing email, or hacking a reputable web site that the victim will visit. The issue can only be resolved on the server side by requiring that all authenticated state-changing requests include an additional piece of secret payload (canary or CSRF token) which is known only to the legitimate web site and the browser and which is protected in transit through SSL/TLS. See the Forgery Protection property on the flow stencil for a list of mitigations.

Justification: Require a state cookie be used and matched against a transmitted state parameter

15. Elevation by Changing the Execution Flow in Keycloak [State: Not Started] [Priority: High]

Category: Elevation Of Privilege

Description: An attacker may pass data into Keycloak in order to change the flow of program execution within Keycloak to the attacker's choosing.

Justification: Utilize ACLs and validate all input

16. Keycloak May be Subject to Elevation of Privilege Using Remote Code Execution [State: Not Started] [Priority: High]

Category: Elevation Of Privilege

Description: End User may be able to remotely execute code for Keycloak.

Justification: Vallidate all input

17. Data Flow Authentication Request Is Potentially Interrupted [State: Not Started] [Priority: High]

Category: Denial Of Service

Description: An external agent interrupts data flowing across a trust boundary in either direction.

Justification: Ability to continue processing in reduced capacity

18. Potential Process Crash or Stop for Keycloak [State: Not Started] [Priority: High]

Category: Denial Of Service
Description: Keycloak crashes, halts, stops or runs slowly; in all cases violating an availability metric.
Justification: Validate all input and ability to continue processing in a reduced capacity.

19. Data Flow Sniffing [State: Not Started] [Priority: High]

Category: Information Disclosure
Description: Data flowing across Authentication Request may be sniffed by an attacker. Depending on what type of data an attacker can read, it may be used to attack other parts of the system or simply be a disclosure of information leading to compliance violations. Consider encrypting the data flow.
Justification: Use strong encryption algorithm

20. Potential Data Repudiation by Keycloak [State: Not Started] [Priority: High]

Category: Repudiation
Description: Keycloak 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.
Justification: Logging and auditing of recieved data.

21. Potential Lack of Input Validation for Keycloak [State: Not Started] [Priority: High]

Category: Tampering
Description: Data flowing across Authentication Request may be tampered with by an attacker. This may lead to a denial of service attack against Keycloak or an elevation of privilege attack against Keycloak or an information disclosure by Keycloak. Failure to verify that input is as expected is a root cause of a very large number of exploitable issues. Consider all paths and the way they handle data. Verify that all input is verified for correctness using an approved list input validation approach.
Justification: Validate all input

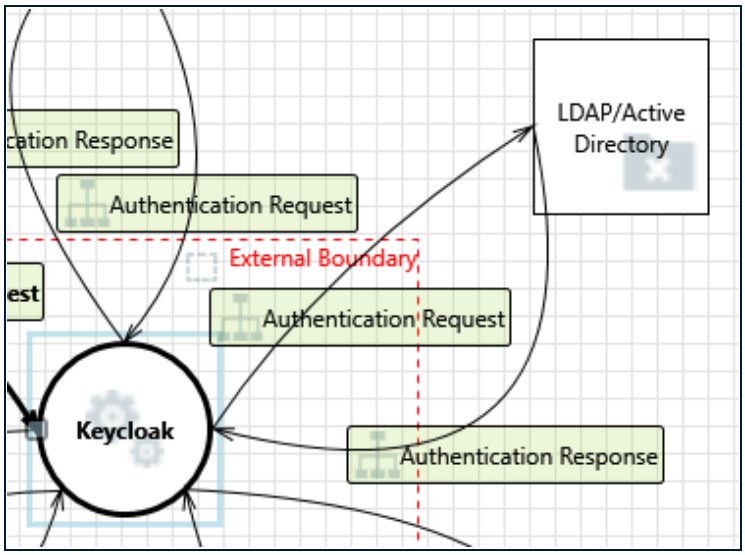
22. Spoofing the Keycloak Process [State: Not Started] [Priority: High]

Category: Spoofing
Description:

Keycloak may be spoofed by an attacker and this may lead to information disclosure by End User. Consider using a standard authentication mechanism to identify the destination process.

Justification: Utilize multi-factor authentication

Interaction: Authentication Request



23. Data Flow Authentication Request Is Potentially Interrupted [State: Not Started] [Priority: High]

Category: Denial Of Service

Description: An external agent interrupts data flowing across a trust boundary in either direction.

Justification: Implement ACLs to protect objects from deletion or modification

24. External Entity LDAP/Active Directory Potentially Denies Receiving Data [State: Not Started] [Priority: High]

Category: Repudiation

Description: LDAP/Active Directory 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.

Justification: LDAP logging and auditing.

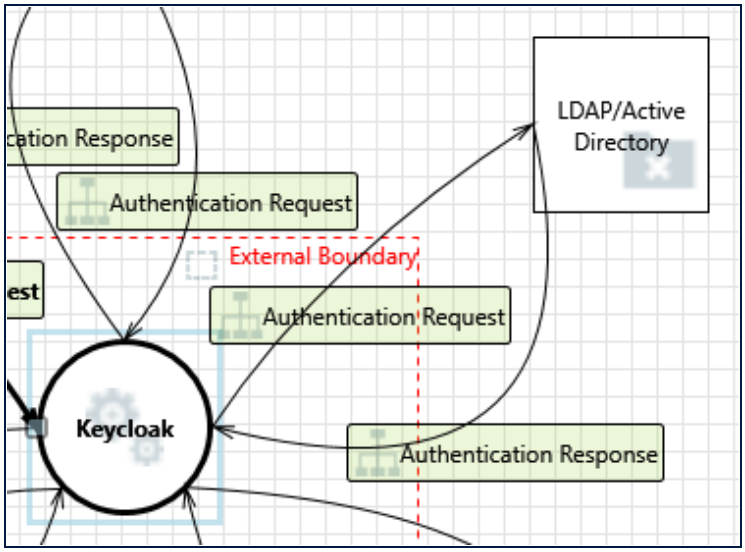
25. Spoofing of the LDAP/Active Directory External Destination Entity [State: Not Started] [Priority: High]

Category: Spoofing

Description: LDAP/Active Directory may be spoofed by an attacker and this may lead to data being sent to the attacker's target instead of LDAP/Active Directory. Consider using a standard authentication mechanism to identify the external entity.

Justification: Use authentication based key exchange

Interaction: Authentication Response



26. Spoofing the LDAP/Active Directory External Entity [State: Not Started] [Priority: High]

Category: Spoofing

Description: LDAP/Active Directory may be spoofed by an attacker and this may lead to unauthorized access to Keycloak. Consider using a standard authentication mechanism to identify the external entity.

Justification: Use authentication based key exchange

27. Elevation Using Impersonation [State: Not Started] [Priority: High]

Category: Elevation Of Privilege

Description: Keycloak may be able to impersonate the context of LDAP/Active Directory in order to gain additional privilege.

Justification: Utilize principle of least privilege

28. Cross Site Request Forgery [State: Not Started] [Priority: High]

Category: Elevation Of Privilege

Description: Cross-site request forgery (CSRF or XSRF) is a type of attack in which an attacker forces a user's browser to make a forged request to a vulnerable site by exploiting an existing trust relationship between the browser and the vulnerable web site. In a simple scenario, a user is logged in to web site A using a cookie as a credential. The other browses to web site B. Web site B returns a page with a hidden form that posts to web site A. Since the browser will carry the user's cookie to web site A, web site B now can take any action on web site A, for example, adding an admin to an account. The attack can be used to exploit any requests that the browser automatically authenticates, e.g. by session cookie, integrated authentication, IP whitelisting, ... The attack can be carried out in many ways such as by luring the victim to a site under control of the attacker, getting the user to click a link in a phishing email, or hacking a reputable web site that the victim will visit. The issue can only be resolved on the server side by requiring that all authenticated state-changing requests include an additional piece of secret payload (canary or CSRF token) which is known only to the legitimate web site and the browser and which is protected in transit through SSL/TLS. See the Forgery Protection property on the flow stencil for a list of mitigations.

Justification: Escape all variables using the right ldap encoding function

29. Elevation by Changing the Execution Flow in Keycloak [State: Not Started] [Priority: High]

Category: Elevation Of Privilege

Description: An attacker may pass data into Keycloak in order to change the flow of program execution within Keycloak to the attacker's choosing.

Justification: Implement ACLs, Input Validation, and setup AD security groups and roles

30. Keycloak May be Subject to Elevation of Privilege Using Remote Code Execution [State: Not Started] [Priority: High]

Category: Elevation Of Privilege

Description: LDAP/Active Directory may be able to remotely execute code for Keycloak.

Justification: Vallidate all input and use security groups and roles

31. Data Flow Authentication Response Is Potentially Interrupted [State: Not Started] [Priority: High]

Category: Denial Of Service

Description: An external agent interrupts data flowing across a trust boundary in either direction.

Justification: Implement ACLs to protect objects from deletion or modification

32. Potential Process Crash or Stop for Keycloak [State: Not Started] [Priority: High]

Category: Denial Of Service

Description: Keycloak crashes, halts, stops or runs slowly; in all cases violating an availability metric.

Justification: Not Applicable (inaccessibility threat does not apply to this process)

33. Data Flow Sniffing [State: Not Started] [Priority: High]

Category: Information Disclosure

Description: Data flowing across Authentication Response may be sniffed by an attacker. Depending on what type of data an attacker can read, it may be used to attack other parts of the system or simply be a disclosure of information leading to compliance violations. Consider encrypting the data flow.

Justification: Use ACLs and encryption protocols

34. Potential Data Repudiation by Keycloak [State: Not Started] [Priority: High]

Category: Repudiation

Description: Keycloak 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.

Justification: LDAP logging and auditing

35. Potential Lack of Input Validation for Keycloak [State: Not Started] [Priority: High]

Category: Tampering

Description: Data flowing across Authentication Response may be tampered with by an attacker. This may lead to a denial of service attack against Keycloak or an elevation of privilege attack against Keycloak or an information disclosure by Keycloak. Failure to verify that input is as expected is a root cause of a very large number of exploitable issues. Consider all paths and the way they handle data. Verify that all input is verified for correctness using an approved list input validation approach.

Justification: Implement ACLs

36. Spoofing the Keycloak Process [State: Not Started] [Priority: High]

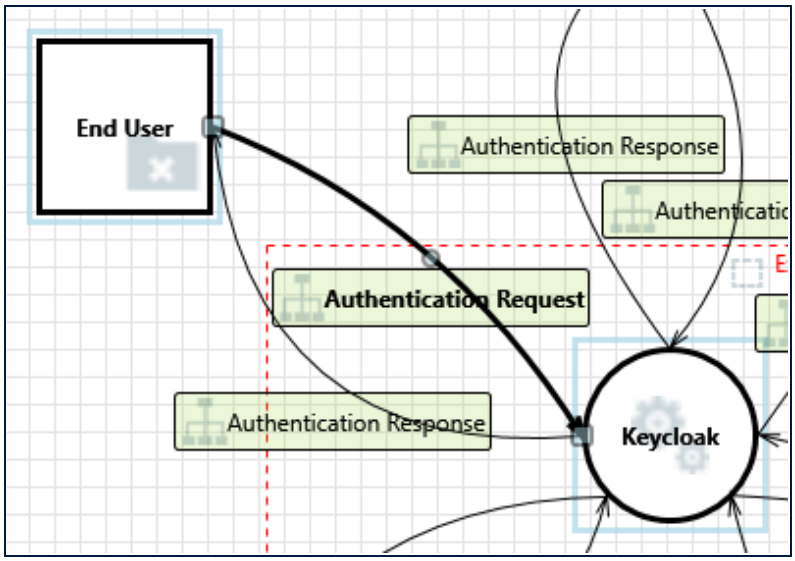
Category: Spoofing

Description:

Keycloak may be spoofed by an attacker and this may lead to information disclosure by LDAP/Active Directory. Consider using a standard authentication mechanism to identify the destination process.

Justification: Use authentication based key exchange

Interaction: Authentication Response



37. Data Flow Authentication Response Is Potentially Interrupted [State: Not Started] [Priority: High]

Category: Denial Of Service

Description: An external agent interrupts data flowing across a trust boundary in either direction.

Justification: Do not block waiting for responses that cross the trust boundary

38. External Entity End User Potentially Denies Receiving Data [State: Not Started] [Priority: High]

Category: Repudiation

Description: End 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.

Justification: Logging and auditing of received data.

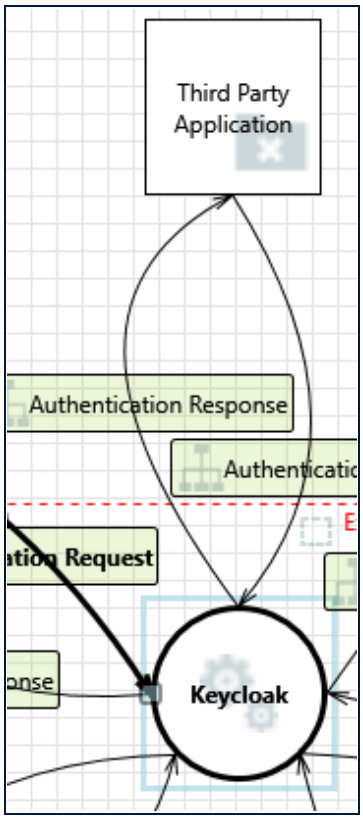
39. Spoofing of the End User External Destination Entity [State: Not Started] [Priority: High]

Category: Spoofing

Description: End User may be spoofed by an attacker and this may lead to data being sent to the attacker's target instead of End User. Consider using a standard authentication mechanism to identify the external entity.

Justification: Utilize multi-factor authentication

Interaction: Authentication Response



40. Data Flow Authentication Response Is Potentially Interrupted [State: Not Started] [Priority: High]

Category: Denial Of Service

Description: An external agent interrupts data flowing across a trust boundary in either direction.

Justification: Do not block waiting for responses that cross the trust boundary

41. External Entity Third Party Application Potentially Denies Receiving Data [State: Not Started] [Priority: High]

Category: Repudiation

Description:

Third Party Application 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.

Justification: Logging and auditing of received data.

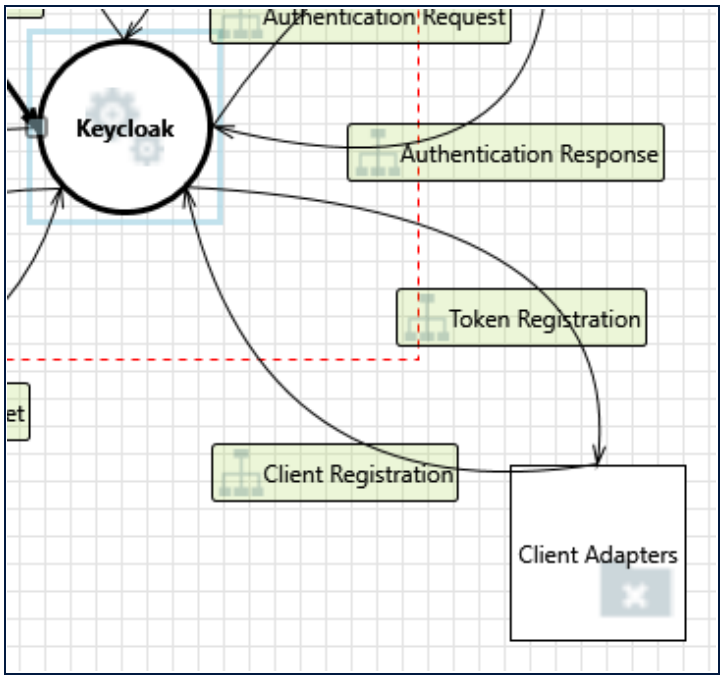
42. Spoofing of the Third Party Application External Destination Entity [State: Not Started] [Priority: High]

Category: Spoofing

Description: Third Party Application may be spoofed by an attacker and this may lead to data being sent to the attacker's target instead of Third Party Application. Consider using a standard authentication mechanism to identify the external entity.

Justification: Utilize multi-factor authentication

Interaction: Client Registration



43. Elevation Using Impersonation [State: Not Started] [Priority: High]

Category: Elevation Of Privilege

Description: Keycloak may be able to impersonate the context of Client Adapters in order to gain additional privilege.

Justification: Utilize principle of least privilege

44. Spoofing the Client Adapters External Entity [State: Not Started] [Priority: High]

Category: Spoofing

Description: Client Adapters may be spoofed by an attacker and this may lead to unauthorized access to Keycloak. Consider using a standard authentication mechanism to identify the external entity.

Justification: Utilize multi-factor authentication

45. Spoofing the Keycloak Process [State: Not Started] [Priority: High]

Category: Spoofing

Description: Keycloak may be spoofed by an attacker and this may lead to information disclosure by Client Adapters. Consider using a standard authentication mechanism to identify the destination process.

Justification: Use multi-factor authentication

46. Potential Lack of Input Validation for Keycloak [State: Not Started] [Priority: High]

Category: Tampering

Description: Data flowing across Client Registration may be tampered with by an attacker. This may lead to a denial of service attack against Keycloak or an elevation of privilege attack against Keycloak or an information disclosure by Keycloak. Failure to verify that input is as expected is a root cause of a very large number of exploitable issues. Consider all paths and the way they handle data. Verify that all input is verified for correctness using an approved list input validation approach.

Justification: Normalization before sanitization

47. Potential Data Repudiation by Keycloak [State: Not Started] [Priority: High]

Category: Repudiation

Description: Keycloak 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.

Justification: Logging and auditing of recieved data.

48. Data Flow Sniffing [State: Not Started] [Priority: High]

Category: Information Disclosure

Description:

Data flowing across Client Registration may be sniffed by an attacker. Depending on what type of data an attacker can read, it may be used to attack other parts of the system or simply be a disclosure of information leading to compliance violations. Consider encrypting the data flow.

Justification: Use encryption with authentication

49. Potential Process Crash or Stop for Keycloak [State: Not Started] [Priority: High]

Category: Denial Of Service

Description: Keycloak crashes, halts, stops or runs slowly; in all cases violating an availability metric.

Justification: Validate all input and ability to continue processing in a reduced capacity.

50. Data Flow Client Registration Is Potentially Interrupted [State: Not Started] [Priority: High]

Category: Denial Of Service

Description: An external agent interrupts data flowing across a trust boundary in either direction.

Justification: Ability to continue processing in reduced capacity

51. Keycloak May be Subject to Elevation of Privilege Using Remote Code Execution [State: Not Started] [Priority: High]

Category: Elevation Of Privilege

Description: Client Adapters may be able to remotely execute code for Keycloak.

Justification: Vallidate all input

52. Elevation by Changing the Execution Flow in Keycloak [State: Not Started] [Priority: High]

Category: Elevation Of Privilege

Description: An attacker may pass data into Keycloak in order to change the flow of program execution within Keycloak to the attacker's choosing.

Justification: Implemented ACLs and Input Validation

53. Cross Site Request Forgery [State: Not Started] [Priority: High]

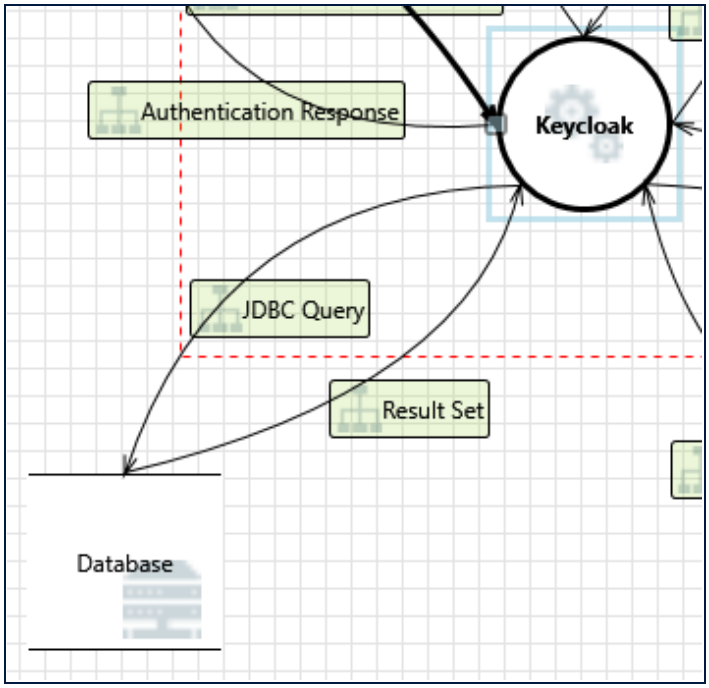
Category: Elevation Of Privilege

Description: Cross-site request forgery (CSRF or XSRF) is a type of attack in which an attacker forces a user's browser to make a forged request to a vulnerable site by exploiting an existing trust relationship between the browser and the vulnerable web site. In a

simple scenario, a user is logged in to web site A using a cookie as a credential. The other browses to web site B. Web site B returns a page with a hidden form that posts to web site A. Since the browser will carry the user's cookie to web site A, web site B now can take any action on web site A, for example, adding an admin to an account. The attack can be used to exploit any requests that the browser automatically authenticates, e.g. by session cookie, integrated authentication, IP whitelisting, ... The attack can be carried out in many ways such as by luring the victim to a site under control of the attacker, getting the user to click a link in a phishing email, or hacking a reputable web site that the victim will visit. The issue can only be resolved on the server side by requiring that all authenticated state-changing requests include an additional piece of secret payload (canary or CSRF token) which is known only to the legitimate web site and the browser and which is protected in transit through SSL/TLS. See the Forgery Protection property on the flow stencil for a list of mitigations.

Justification: Require a state cookie be used and matched against a transmitted state parameter

Interaction: JDBC Query



54. Spoofing of Destination Data Store Database [State: Not Started] [Priority: High]

Category: Spoofing

Description: Database may be spoofed by an attacker and this may lead to data being written to the attacker's target instead of Database. Consider using a standard authentication mechanism to identify the destination data store.

Justification: Use authentication based key exchange

55. Potential Excessive Resource Consumption for Keycloak or Database [State: Not Started] [Priority: High]

Category: Denial Of Service

Description: Does Keycloak or 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.

Justification: Configuration management on the data store to set time out.

56. Spoofing the Keycloak Process [State: Not Started] [Priority: High]

Category: Spoofing

Description: Keycloak may be spoofed by an attacker and this may lead to unauthorized access to Database. Consider using a standard authentication mechanism to identify the source process.

Justification: Use authentication based key exchange

57. The Database Data Store Could Be Corrupted [State: Not Started] [Priority: High]

Category: Tampering

Description: Data flowing across JDBC Query may be tampered with by an attacker. This may lead to corruption of Database. Ensure the integrity of the data flow to the data store.

Justification: Normalization before being set along with authentication

58. Data Store Denies Database Potentially Writing Data [State: Not Started] [Priority: High]

Category: Repudiation

Description: Database claims that it did not write data received from an entity on the other side of the trust boundary. Consider using logging or auditing to record the source, time, and summary of the received data.

Justification: Logging on both process side and data store side

59. Data Flow Sniffing [State: Not Started] [Priority: High]

Category: Information Disclosure

Description:

Data flowing across JDBC Query may be sniffed by an attacker. Depending on what type of data an attacker can read, it may be used to attack other parts of the system or simply be a disclosure of information leading to compliance violations. Consider encrypting the data flow.

Justification: Use encryption with authentication

60. Data Flow JDBC Query Is Potentially Interrupted [State: Not Started] [Priority: High]

Category: Denial Of Service

Description: An external agent interrupts data flowing across a trust boundary in either direction.

Justification: Handling all possible exceptions in a manner which prevents application down time.

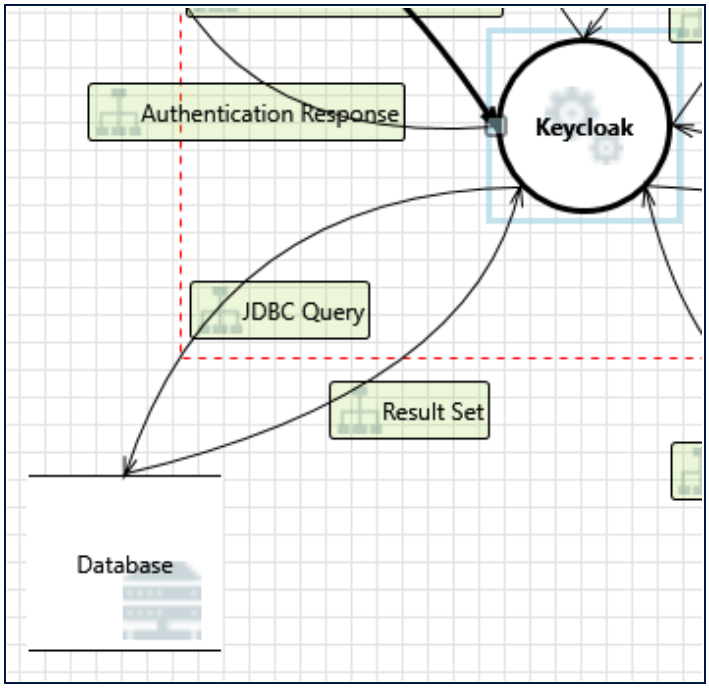
61. Data Store Inaccessible [State: Not Started] [Priority: High]

Category: Denial Of Service

Description: An external agent prevents access to a data store on the other side of the trust boundary.

Justification: Not applicable (inaccessible threat does not apply to this data store)

Interaction: Result Set



62. Spoofing of Source Data Store Database [State: Not Started] [Priority: High]

Category: Spoofing

Description: Database may be spoofed by an attacker and this may lead to incorrect data delivered to Keycloak. Consider using a standard authentication mechanism to identify the source data store.

Justification: Use authentication based key exchange

63. Weak Access Control for a Resource [State: Not Started] [Priority: High]

Category: Information Disclosure

Description: Improper data protection of Database can allow an attacker to read information not intended for disclosure. Review authorization settings.

Justification: Use encryption with authentication

64. Spoofing the Keycloak Process [State: Not Started] [Priority: High]

Category: Spoofing

Description: Keycloak may be spoofed by an attacker and this may lead to information disclosure by Database. Consider using a standard authentication mechanism to identify the destination process.

Justification: Use authentication based key exchange

65. Potential Data Repudiation by Keycloak [State: Not Started] [Priority: High]

Category: Repudiation

Description: Keycloak 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.

Justification: Logging on both process side and data store side.

66. Potential Process Crash or Stop for Keycloak [State: Not Started] [Priority: High]

Category: Denial Of Service

Description: Keycloak crashes, halts, stops or runs slowly; in all cases violating an availability metric.

Justification: Not Applicable (inaccessibility threat does not apply to this process)

67. Data Flow Result Set Is Potentially Interrupted [State: Not Started] [Priority: High]

Category: Denial Of Service

Description: An external agent interrupts data flowing across a trust boundary in either direction.

Justification: Handling all possible exceptions in a manner which prevents application down time.

68. Data Store Inaccessible [State: Not Started] [Priority: High]

Category: Denial Of Service

Description: An external agent prevents access to a data store on the other side of the trust boundary.

Justification: Not applicable (inaccessible threat does not apply to this data store)

69. Keycloak May be Subject to Elevation of Privilege Using Remote Code Execution [State: Not Started] [Priority: High]

Category: Elevation Of Privilege

Description: Database may be able to remotely execute code for Keycloak.

Justification: Implemented ACLs and Input Validation from the data store

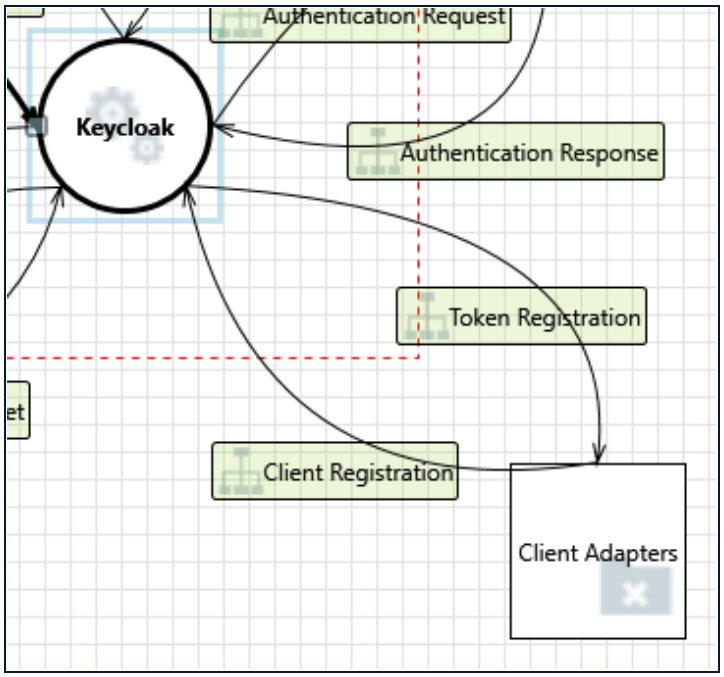
70. Elevation by Changing the Execution Flow in Keycloak [State: Not Started] [Priority: High]

Category: Elevation Of Privilege

Description: An attacker may pass data into Keycloak in order to change the flow of program execution within Keycloak to the attacker's choosing.

Justification: Implemented ACLs and Input Validation from the data store

Interaction: Token Registration



71. Spoofing of the Client Adapters External Destination Entity [State: Not Started] [Priority: High]

Category: Spoofing

Description: Client Adapters may be spoofed by an attacker and this may lead to data being sent to the attacker's target instead of Client Adapters. Consider using a standard authentication mechanism to identify the external entity.

Justification: Utilize multi-factor authentication

72. External Entity Client Adapters Potentially Denies Receiving Data [State: Not Started] [Priority: High]

Category: Repudiation

Description: Client Adapters 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.

Justification: Logging and auditing of received data.

73. Data Flow Token Registration Is Potentially Interrupted [State: Not Started] [Priority: High]

Category: Denial Of Service

Description: An external agent interrupts data flowing across a trust boundary in either direction.

Justification: Do not block waiting for responses that cross the trust boundary