Threat Modeling Report

Created on 11/12/2019 6:30:17 PM

Threat Model Name:

Owner:

Reviewer:

Contributors:

Description:

Assumptions:

External Dependencies:

Threat Model Summary:

Not Started	0
Not Applicable	12
Needs Investigation	2
Mitigation Implemented	10
Total	24
Total Migrated	0

Diagram: Diagram 1

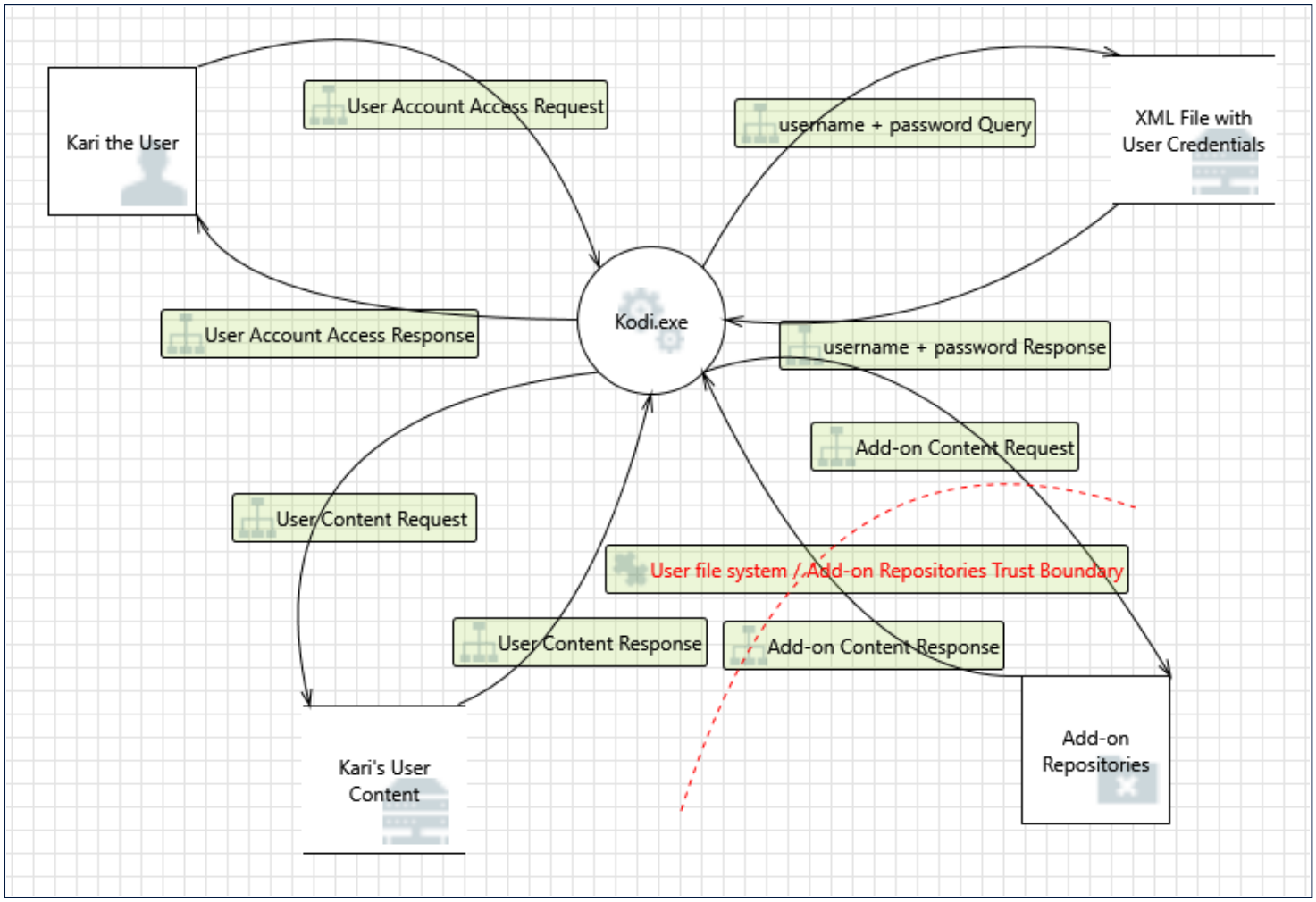
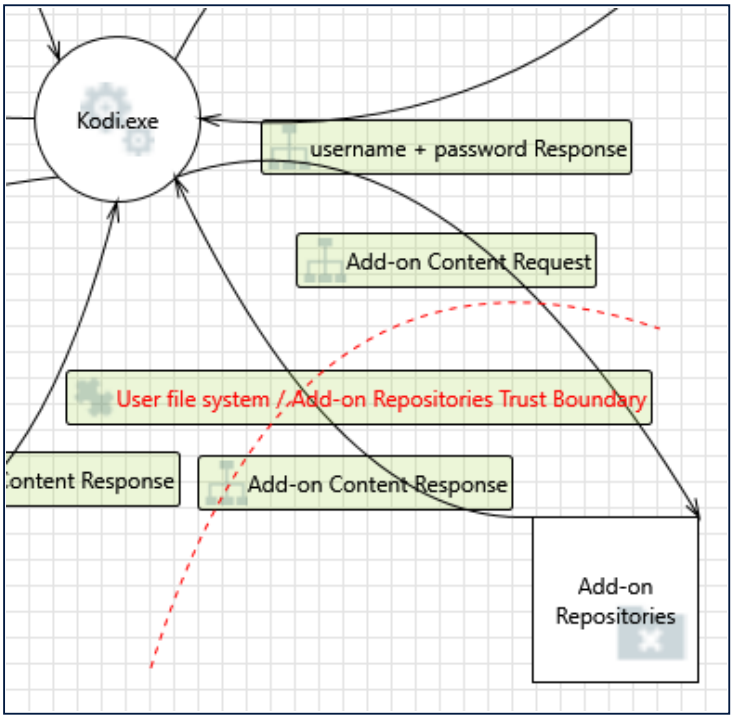


Diagram 1 Diagram Summary:

Not Started	0
Not Applicable	12
Needs Investigation	2
Mitigation Implemented	10
Total	24
Total Migrated	0

Interaction: Add-on Content Request



1. External Entity Add-on Repositories Potentially Denies Receiving Data [State: Not Applicable] [Priority: High]

Category: Repudiation

Description: Add-on Repositories 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: After initial testing, Kodi only becomes aware of issues with Add-ons when users report it, only then will it be known to Kodi and investigated

2. Spoofing of the Add-on Repositories External Destination Entity [State: Mitigation Implemented] [Priority: High]

Category: Spoofing

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

Justification: Using HTTPS to encrypt traffic will help mitigate spoofing as well as data source validation and code integrety check.

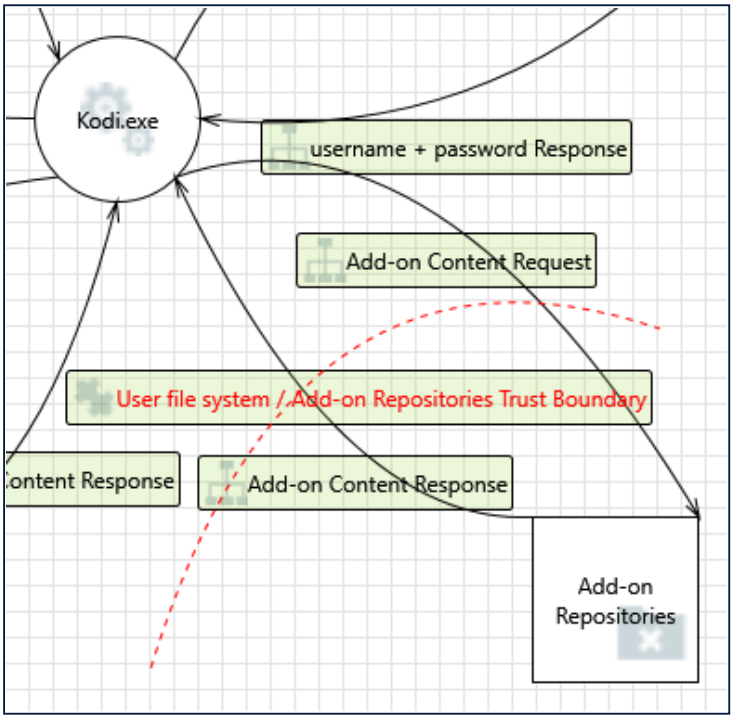
3. Data Flow Add-on Content Request Is Potentially Interrupted [State: Not Applicable] [Priority: High]

Category: Denial Of Service

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

Justification: After initial testing, Kodi only becomes aware of issues with Add-ons when users report it, only then will it be known to Kodi and investigated

Interaction: Add-on Content Response



4. Elevation Using Impersonation [State: Not Applicable] [Priority: High]

Category: Elevation Of Privilege

Description: Kodi.exe may be able to impersonate the context of Add-on Repositories in order to gain additional privilege.

Justification: User to notify Kodi of any problems per statement in GUI

5. Spoofing the Generic External Interactor External Entity [State: Mitigation Implemented] [Priority: High]

Category: Spoofing

Description: Add-on Repositories may be spoofed by an attacker and this may lead to unauthorized access to Kodi.exe. Consider using a standard authentication mechanism to identify the external entity.

Justification: Using HTTPS to encrypt traffic will help mitigate spoofing as well as input and data source validation

6. Spoofing the Kodi.exe Process [State: Mitigation Implemented] [Priority: High]

Category: Spoofing

Description: Kodi.exe may be spoofed by an attacker and this may lead to information disclosure by Add-on Repositories. Consider using a standard authentication mechanism to identify the destination process.

Justification: Using HTTPS to encrypt traffic will help mitigate spoofing as well as input and data source validation

7. Potential Lack of Input Validation for Kodi.exe [State: Not Applicable] [Priority: High]

Category: Tampering

Description: Data flowing across Add-on Content Response may be tampered with by an attacker. This may lead to a denial of service attack against Kodi.exe or an elevation of privilege attack against Kodi.exe or an information disclosure by Kodi.exe. 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: After initial testing, Kodi only becomes aware of issues with Add-ons when users report it, only then will it be known to Kodi and investigated

8. Potential Data Repudiation by Kodi.exe [State: Not Applicable] [Priority: High]

Category: Repudiation

Description: Kodi.exe 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: After initial testing, Kodi only becomes aware of issues with Add-ons when users report it, only then will it be known to Kodi and investigated

9. Data Flow Sniffing [State: Mitigation Implemented] [Priority: High]

Category: Information Disclosure

Description: Data flowing across Add-on Content 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: Using HTTPS to access Repositories will help mitigate information disclosure.

10. Potential Process Crash or Stop for Kodi.exe [State: Not Applicable] [Priority: High]

Category: Denial Of Service

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

Justification: User to notify Kodi of any problems per statement in GUI

11. Data Flow Add-on Content Response Is Potentially Interrupted [State: Not Applicable] [Priority: High]

Category: Denial Of Service

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

Justification: User to notify Kodi of any problems per statement in GUI

12. Kodi.exe May be Subject to Elevation of Privilege Using Remote Code Execution [State: Needs Investigation] [Priority: High]

Category: Elevation Of Privilege

Description: Add-on Repositories may be able to remotely execute code for Kodi.exe.

Justification: This is part of the process of Kodi. Kodi runs as its own executable and when an Add-on is accessed it opens up its own executable on the OS

13. Elevation by Changing the Execution Flow in Kodi.exe [State: Mitigation Implemented] [Priority: High]

Category: Elevation Of Privilege

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

Justification: The user can set up a Master Lock PIN to mitigate the potential elevation of privilege.

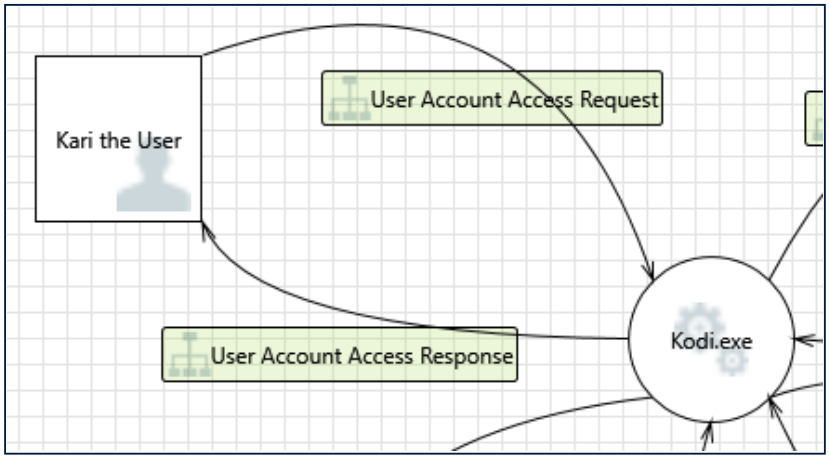
14. Cross Site Request Forgery [State: Needs Investigation] [Priority: High]

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: It is unclear how Kodi and the Add-ons perform authentication. A checksum is used to see if there is a change in content however this change is not guaranteed to be a legitimate update or malicious change. Kodi also makes this connection via HTTP not HTTPS

Interaction: User Account Access Request



15. Elevation Using Impersonation [State: Not Applicable] [Priority: High]

Category: Elevation Of Privilege

Description: Kodi.exe may be able to impersonate the context of Kari the User in order to gain additional privilege.

Justification: The user can set up a Master Lock PIN to mitigate the potential elevation of privilege.

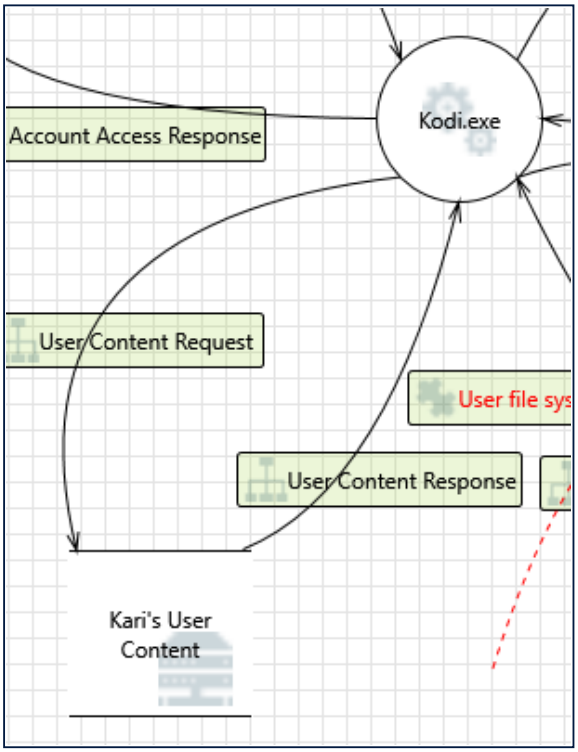
16. Spoofing the Kari the User External Entity [State: Mitigation Implemented] [Priority: High]

Category: Spoofing

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

Justification: The user can set up a Master Lock PIN to produce an authentication method.

Interaction: User Content Request



17. Potential Excessive Resource Consumption for Kodi.exe or Kari's User Content [State: Not Applicable] [Priority: High]

Category: Denial Of Service

Description: Does Kodi.exe or Kari's User Content 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: Files and executable are stored on user's hardware and filesystem therefor resource consumption should be implemented by the user's OS. This is out of scope for the process.

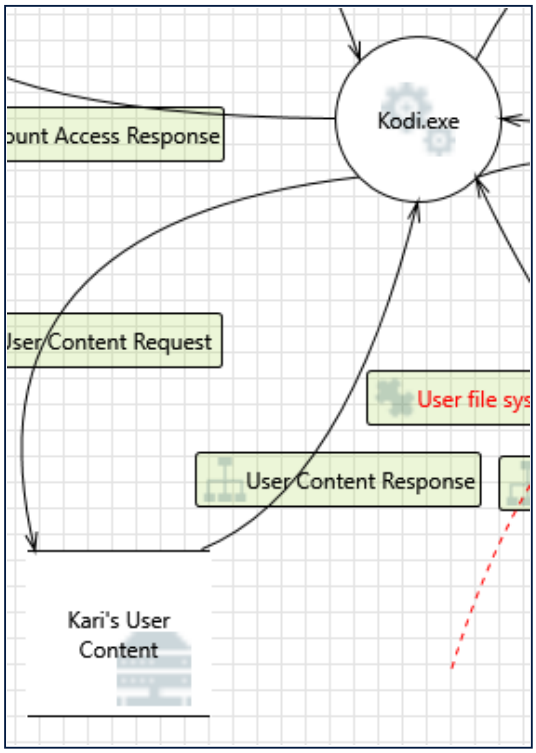
18. Spoofing of Destination Data Store Kari's User Content [State: Mitigation Implemented] [Priority: High]

Category: Spoofing

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

Justification: The user can set up a Master Lock PIN for authentication.

Interaction: User Content Response



19. Weak Access Control for a Resource [State: Mitigation Implemented] [Priority: High]

Category: Information Disclosure

Description: Improper data protection of Kari's User Content can allow an attacker to read information not intended for disclosure. Review authorization settings.

Justification: Notify user that data is stored insecurely to and implement their own data protection.

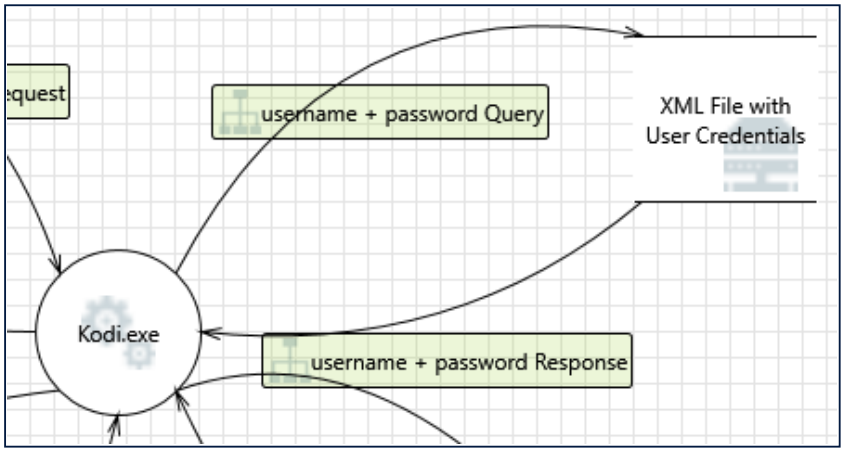
20. Spoofing of Source Data Store Kari's User Content [State: Not Applicable] [Priority: High]

Category: Spoofing

Description: Kari's User Content may be spoofed by an attacker and this may lead to incorrect data delivered to Kodi.exe. Consider using a standard authentication mechanism to identify the source data store.

Justification: The file is stored locally on the user's hardware and filesystem. The hardware and filesystem are outside of the scope of the process

Interaction: username + password Query



21. Potential Excessive Resource Consumption for Kodi.exe or User Credentials File [State: Not Applicable] [Priority: High]

Category: Denial Of Service

Description: Does Kodi.exe or XML File with User Credentials 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: Files and executable are stored on user's hardware and filesystem therefor resource consumption should be implemented by the user's OS.

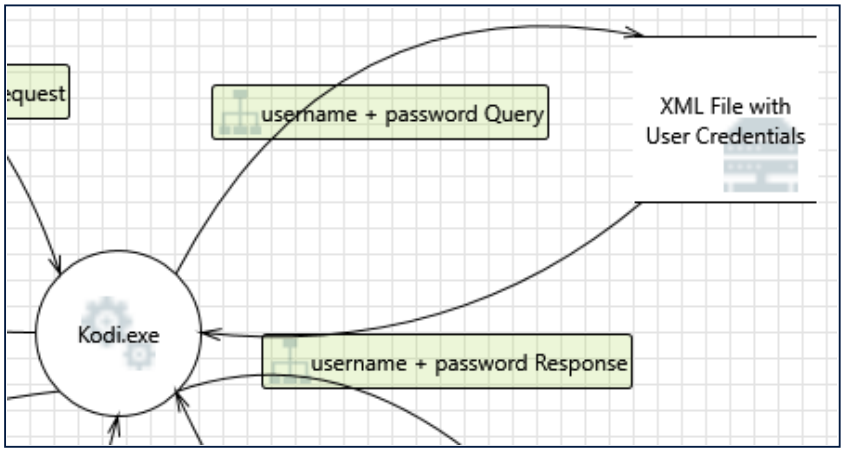
22. Spoofing of Destination Data Store User Credentials File [State: Mitigation Implemented] [Priority: High]

Category: Spoofing

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

Justification: The user can set up a Master Lock PIN to produce an authentication method.

Interaction: username + password Response



23. Weak Access Control for a Resource [State: Mitigation Implemented] [Priority: High]

Category: Information Disclosure

Description: Improper data protection of XML File with User Credentials can allow an attacker to read information not intended for disclosure. Review authorization settings.

Justification: Notify user that usernames and passwords are stored in plaintext and to implement their own data protection or secondary authentication

24. Spoofing of Source Data Store User Credentials File [State: Not Applicable] [Priority: High]

Category: Spoofing

Description: XML File with User Credentials may be spoofed by an attacker and this may lead to incorrect data delivered to Kodi.exe. Consider using a standard authentication mechanism to identify the source data store.

Justification: The XML file is stored locally on the user's hardware and filesystem. The hardware and filesystem are outside of the scope of the process