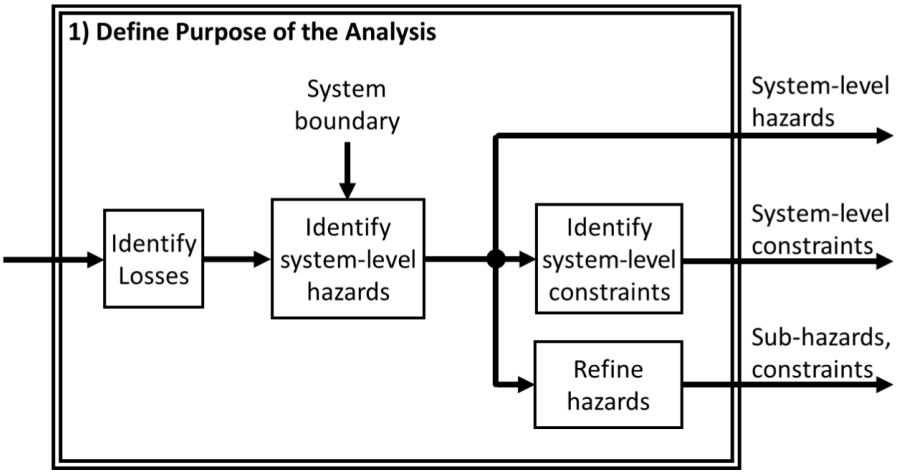
STPA ANALYSIS FOR A ROLE-BASED ACCESS CONTROL (RBAC) SUBSYSTEM

Last update: 06/10/2024

1. Define the purpose of the analysis (STPA Step 1)



1.1 Define and frame the problem

In modern software systems, particularly those handling sensitive information or critical functionalities, authentication and authorization mechanisms are paramount. These mechanisms ensure that only authenticated users with appropriate permissions can access specific resources or perform certain actions within the system. One common approach to implementing authentication and authorization is through Role-Based Access Control (RBAC), where users are assigned roles, and access rights are granted based on those roles.

RBAC is a widely used method for managing access to resources within an organization's information system. RBAC assigns permissions to users based on their roles, rather than specifying individual permissions for each user. This approach o offers several advantages, including improved security and easier management of access rights.

1.2 Stakeholders

S1: **Users**

- Individuals who interact with the web application and require access to its features and resources. Users may include employees, customers, clients, administrators, or any other individuals authorized to use the system.

S2: **Administrators**:

- Responsible for managing the RBAC subsystem within the web application. They create, update, and delete user accounts, roles, and permissions. Administrators also handle user support and troubleshooting related to access issues.

S3: **Developers**

- Responsible for designing, implementing, and maintaining the RBAC subsystem within the web application. They ensure that the system functions correctly, securely, and efficiently, and they may also be involved in integrating RBAC with other parts of the application.

S4: **Management**:

- Oversee the overall operation and strategy of the web application. They may set policies and guidelines related to access control, define business requirements for RBAC implementation, and allocate resources for its development and maintenance.

S5. **Security Personnel**

- Concerned with ensuring the confidentiality, integrity, and availability of data within the web application. They may provide input on security requirements, conduct risk assessments, and monitor the RBAC subsystem for potential vulnerabilities or threats.

S6. **Regulatory Authorities**

- Depending on the industry and jurisdiction, regulatory authorities may have requirements or standards that the web application's RBAC subsystem must comply with. These stakeholders may provide guidelines or regulations related to data privacy, security, and access control.

S7. **Third-party Service Providers**:

- If the web application relies on third-party services or components for authentication or authorization, these providers become stakeholders as well. They may include identity providers, authentication services, or cloud service providers offering RBAC solutions.

S8. **End Users' Organizations**

- In cases where the web application serves organizations rather than individual users, the organizations themselves become stakeholders. They may have specific requirements for user management, access control, and data security within the application.

S9. **Auditors**

- Auditors may include internal auditors employed by the organization itself, external auditors hired by regulatory bodies or independent firms, or compliance officers responsible for ensuring adherence to industry standards and best practices. Auditors play a crucial role in evaluating the effectiveness of the RBAC subsystem within the web application by conducting audits, inspections, or assessments to identify any deficiencies, weaknesses, or non-compliance issues. They provide independent oversight and assurance to stakeholders regarding the system's security, integrity, and regulatory compliance.

1.3 System purpose and goal

SG-01: Ensure Authentication

- The system should accurately verify the identity of users seeking access to resources or functionalities.

SG-02: Enforce Authorization

- The system should grant access rights based on predefined roles and permissions, ensuring that users can only perform actions they are authorized to do.

SG-03: Maintain Data Confidentiality

- RBAC helps enforce confidentiality by ensuring that users only have access to the information and resources that are necessary for their roles.

SG-04: Maintain Data Integrity

- RBAC contributes to maintaining data integrity by preventing unauthorized users from making unauthorized changes to information.

SG-05: Facilitate Role Management

- The system should provide administrators with tools to efficiently manage roles, including creating, updating, and deleting roles as needed.

SG-06: Ensure Availability

- RBAC contributes to availability by ensuring that resources are accessible to authorized users when needed.

SG-07: Ensure Auditability

- The system should log user access and actions for auditing purposes, helping to trace any security breaches or unauthorized activities.

1.4 Assumptions

AS-1: System Boundaries

- The RBAC system can be reused as a component in a web application. The RBAC system is responsible for managing user authentication, authorization, and access control within the application's defined boundaries.

AS-2: System Architecture

- The RBAC system is assumed to be designed as a modular component, capable of integrating seamlessly with an existing architecture of a web application. It interacts with other components such as user interfaces, databases, and application logic to enforce access control policies.

AS-3: Web Environment

- The RBAC system operates within a typical web application environment, with assumptions about factors such as network connectivity, server reliability, database availability, and web browser compatibility.

AS-4: Roles

- Users interacting with the web application are assumed to have varying roles, permissions, and access levels determined by the RBAC system.

AS-5: Regulatory and Legal compliance

- The RBAC system is assumed to comply with relevant laws, regulations, and industry standards governing data privacy, security, and access control.

AS-6: Scalability and Performance

- The RBAC system is assumed to be capable of accommodating a growing number of users, roles, and permissions as the web application scales. It should also be optimized for performance to handle concurrent user requests efficiently without compromising security.

1.5 Identify losses and accidents

L1: User cannot log in
L2: Disclosure of sensitive user data.
L3: User account compromised due to third-party intrusion.
L4: Unauthorized user gains access to confidential information.
L5: System inability to recover user data.
L6: User receives incorrect role from the Authorizer.
L7: User incorrectly authenticated by the Authenticator.

1.6 Identify system hazards and system level constraints

1.6.1. Identify system hazards

H1: System overload. **[L1, L5]**
H2: Absence of password security policies. **[L2, L3]**
H3: No role assigned to an authenticated user. **[L4, L6]**
H4: No validation of credentials before authorization. **[L2, L3, L4, L6, L7]**
H5: Lack of data protection. **[L2, L3, L5]**
H6: User retains access after permissions are revoked. **[L2, L3, L4]**

1.6.2. Identify system constraints

SC1: The system must not be overloaded. **[H1]**
SC2: Password security policy must be in place. **[H2]**
SC3: The Authenticator must associate a role with an authenticated user. **[H3]**
SC4: Credentials must be validated before authorization. **[H4]**
SC5: The repository must have protection. **[H5]**
SC6: Users must lose access after permissions are revoked. **[H6]**
SC7: Only authenticated users are able to access the system. **[H2] [H4]**

2. Model the Control Structure (STPA Step 2)

2.1 System Components

- Usuário (User)
- Controlador de login (Login controller)
- Autenticador (Authenticator)
- Repositório (Repository)

2.2 Component responsibilities

User		
R1	Manter suas credenciais (`username` e `password`). As credenciais podem, por exemplo, serem salvas em um arquivo, anotadas em um caderno, ou até mesmo memorizadas.	SC2 SC7
R2	Fornecer suas credenciais quando solicitado.	SC2 SC7
R3	Assegurar que a senha associada ao usuário seja forte de acordo com a política de segurança da organização.	SC2

Login Controller		
R1	Tratar requisições de autenticação	SC7
R2	Gerenciar usuários	SC7

Authenticator		
R1	Validar credenciais fornecidas pelo usuário com credenciais armazenadas no repositório.	SC4
R2	Trabalhar em conjunto com o Autorizador para correta associação de papéis a usuários autenticados.	SC3
R3	Assegurar que o papel(is) adequado(s) seja(m) atribuído(s) a todo usuário autenticado.	SC3

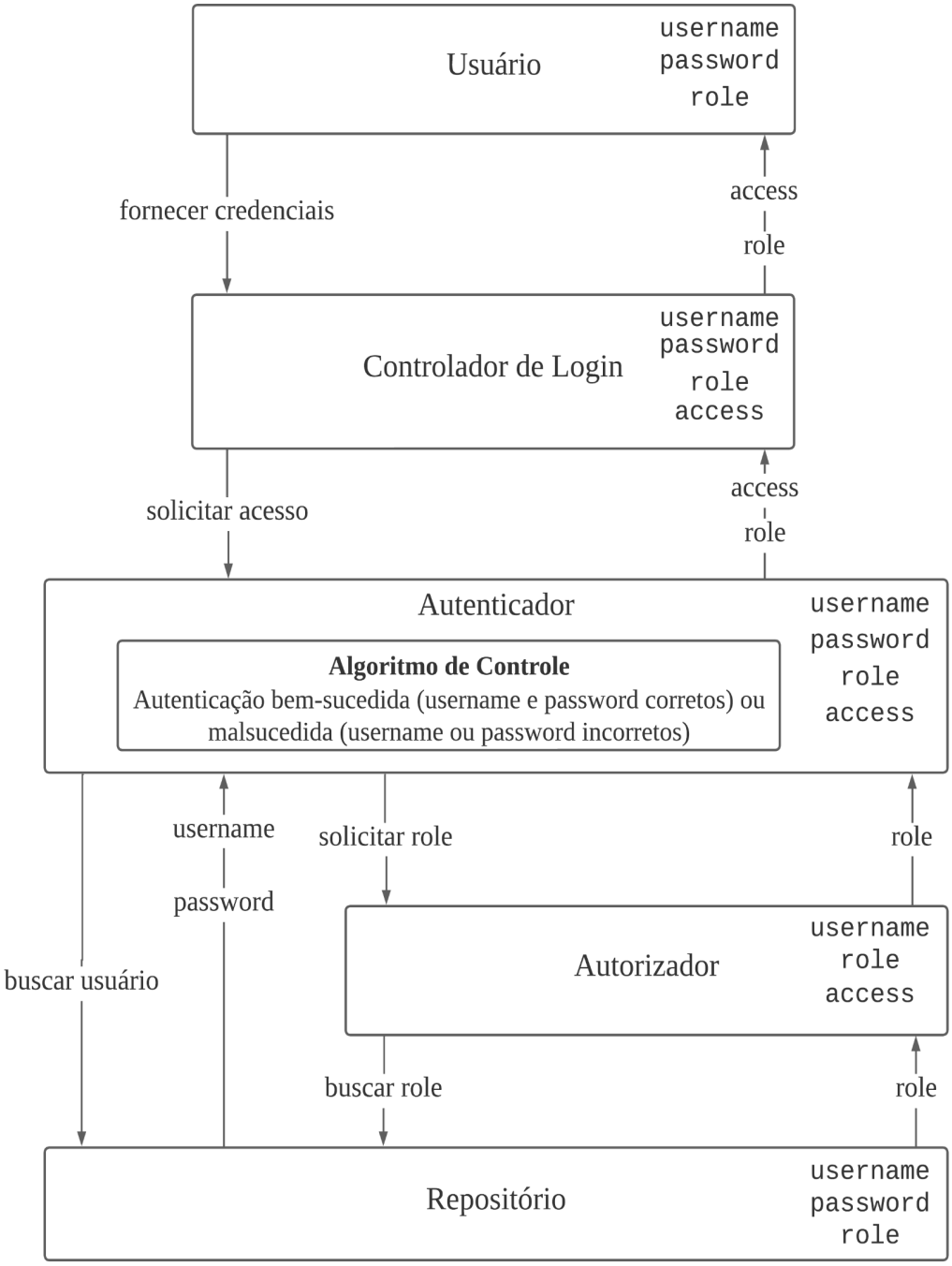
Autorizador		
R1	Definir o papel adequado para o usuário autenticado	SC3
R2	Gerenciar papéis e permissões	SC3 SC7

Repository		
R1	Manter as credenciais dos usuários com seus respectivos papéis.	SC5

2.3 System Level Control Structure

[Represent the System level control structure. Guideline: at this level try to use at most 3 components at a higher level of abstraction. The Control Actions and Feedback between components in the control structure are derived from the Responsibilities previously identified (and, hence, derived also from the System Safety Constraints)].

2.4 Control structure (Refinement/Zoom of the System Level control structure to be analyzed)



Na estrutura de controle temos um processo controlado (**Repositório**) e quatro controladores: **Usuário**, **Controlador de *Login***, **Autenticador** e **Autorizador**.

O controlador **Usuário** é responsável por emitir a *control action* "fornecer credenciais" para tentar obter acesso ao sistema. Para isso, ele envia as variáveis "`username`" e "`password`" para validação no **Repositório**.

O **Controlador de Login** é responsável por intermediar o acesso entre o controlador **Usuário** e o controlador **Autenticador**, através da *control action* "solicitar acesso".

O controlador **Autenticador** é responsável por realizar a autenticação através da *control action* "buscar usuário" (validação do conjunto "`username`" e "`password`" no **Repositório**).

Caso o conjunto de "`username`" e "`password`" fornecidos não sejam encontrados no **Repositório**, o **Autenticador** devolve ao **Controlador de Login** o "`access`" com valor "`Not Allowed`" e o "`role`" com valor "`Unknown`".

Caso o conjunto "`username`" e "`password`" sejam encontrados no **Repositório**, o **Autenticador** emite a *control action* "solicitar role" ao **Autorizador**, que por sua vez fornece a *control action* "buscar role" ao **Repositório**.

A resposta do **Repositório** é o *role* encontrado para o "`username`" fornecido. O **Autorizador** devolve o conteúdo da variável "`role`" ao **Autenticador** que por sua vez conclui a autorização, enviando ao **Controlador de Login** e **Usuário** a concessão do acesso (quando "`username`" é "`Valid`", "`password`" é "`Valid`" e "`role`" é "`Admin`" ou "`User`") ou negando acesso (quando "`username`" é "`Valid`", "`password`" é "`Valid`" e "`role`" é "`Unknown`").

Usuário:

Variáveis	Valores
username	[Valid / Not Valid / Provided / Not Provided]
password	[Valid / Not Valid / Provided / Not Provided]
role	[Admin / User / Unknown]

Controlador de Login:

Variáveis	Valores
username	[Valid / Not Valid / Provided / Not Provided]
password	[Valid / Not Valid / Provided / Not Provided]
role	[Admin / User / Unknown]
access	[Allowed / Not Allowed]

Autenticador:

Variáveis	Valores
username	[Valid / Not Valid / Provided / Not Provided]
password	[Valid / Not Valid / Provided / Not Provided]
role	[Admin / User / Unknown]
access	[Allowed / Not Allowed]

Autorizador:

Variáveis	Valores
username	[Valid / Not Valid / Provided / Not Provided]
role	[Admin / User/ Unknown]
access	[Allowed / Not Allowed]

Repositório:

Variáveis	Valores
username	[Valid / Not Valid / Provided / Not Provided]
password	[Valid / Not Valid / Provided / Not Provided]
role	[Admin / User / Unknown]

3. Identify Unsafe Control Actions (STPA Step 3)

3.1 Identify Unsafe Control Actions – UCAs

USUÁRIO				
Control action	**Not providing causes hazard**	**Providing causes hazard**	**Too early, too late, out of order**	**Stopped too soon/Applied tool long**
Fornecer credenciais	**UCA-1:** Usuário não fornece fornecer credenciais quando o nome de usuário é fornecido, a senha é fornecida **[H-1]** **UCA-2:** Usuário não fornece fornecer credenciais quando o nome de usuário não foi fornecido, a senha foi fornecida **[H-1]** **UCA-3:** Usuário não fornece fornecer credenciais quando o nome de usuário é fornecido, a senha não é fornecida **[H-1]**	Sem Perigo	Sem Perigo	N/A

CONTROLADOR DE LOGIN				
Control action	**Not providing causes hazard**	**Providing causes hazard**	**Too early, too late, out of order**	**Stopped too soon/Applied tool long**
Solicitar acesso	**UCA- 4:** Controlador de Login não fornece solicitar acesso quando o nome de usuário é fornecido, a senha é fornecida **[H-1]** **UCA- 5:** Controlador de Login não fornece solicitar acesso quando o nome de usuário não é fornecido, a senha é fornecida **[H-1]** **UCA- 6:** Controlador de Login não fornece solicitar acesso quando nome de usuário não é fornecido, senha não é fornecida **[H-1]** **UCA- 7:** Controlador de Login não fornece solicitar acesso quando o nome de usuário é fornecido, a senha não é fornecida **[H-1]**	Sem Perigo	Sem Perigo	N/A

AUTENTICADOR				
Control action	**Not providing causes hazard**	**Providing causes hazard**	**Too early, too late, out of order**	**Stopped too soon/Applied tool long**
Buscar usuário	**UCA-8:** Autenticador não fornece buscar usuário quando o nome de usuário e a senha são fornecidos. **[H4]** **UCA-9:** Autenticador não fornece buscar usuário quando o nome de usuário não é fornecido e a senha é fornecida. **[H4]** **UCA-10:** Autenticador não fornece buscar usuário quando o nome de usuário é fornecido e a senha não é fornecida. **[H4]** **UCA-11:** Autenticador não fornece para buscar usuário quando o nome de usuário e a senha não são fornecidos. **[H4]**	Sem Perigo	**UCA-12:** Autenticador fornece buscar usuário muito cedo quando o nome de usuário não é fornecido e a senha não é fornecido. **[H2]**	N/A
Solicitar role	**UCA-13:** Autenticador não fornece solicitar role quando o nome de usuário é válido e a senha é válida. **[H1]**	**UCA-14:** Autenticador fornece solicitar role quando o nome de usuário não é válido e a senha é válida. **[H2, H4]**	**UCA-17:** Autenticador fornece solicitar role muito cedo quando o nome de usuário é fornecido e senha é fornecido. **[H2, H4]**	N/A

		UCA-15: Autenticador fornece solicitar role quando o nome de usuário é válido e a senha não é válida. **[H2, H4]** **UCA-16:** Autenticador fornece solicitar role quando o nome de usuário e a senha não são válidos**. [H2, H4]**	**UCA-18:** Autenticador fornece solicitar role muito cedo quando o nome de usuário foi fornecido e a senha não é fornecida. **[H2, H4]** **UCA-19:** Autenticador fornece solicitar role muito cedo quando o nome de usuário não é fornecido e a senha é fornecida. **[H2, H4]** **UCA-20:** Autenticador fornece solicitar role muito cedo quando o nome de usuário não é fornecido e a senha não é fornecido. **[H2, H4]**	

AUTORIZADOR				
Control action	**Not providing causes hazard**	**Providing causes hazard**	**Too early, too late, out of order**	**Stopped too soon/Applied tool long**
Buscar role	**UCA-21:** Autorizador não fornece buscar role quando o nome de usuário é válido e senha é válida. **[H1, H6]**	**UCA-22:** Autorizador fornece buscar role quando nome de usuário não é válido e senha é válida.**[H2, H3, H4, H6]** **UCA-23:** Autorizador fornece buscar role quando nome de usuário é	**UCA-25:** Autorizador fornece buscar role muito cedo quando nome de usuário é fornecido e senha é fornecido. **[H4, H6]** **UCA-26:** Autorizador fornece buscar role muito cedo quando nome de	N/A

		válido e senha não é válida. **[H2, H3, H4, H6]** **UCA-24:** Autorizador fornece buscar role quando nome de usuário não é válido e senha não é válido.**[H2, H3, H4, H6]**	usuário não é fornecido e senha é fornecido. **[H4, H6]** **UCA-27:** Autorizador fornece buscar role muito cedo quando nome de usuário é fornecido e password não é fornecido. **[H4, H6]** **UCA-28:** Autorizador fornece buscar role muito cedo quando nome de usuário não é fornecido e senha não é fornecido. **[H4, H6]**	

3.2 Identify controller constraints

Unsafe Control Actions	**Controller constraint(s)**
UCA-1	C-1, C-2
UCA-2	C-2
UCA…	C..
UCA-n	C-n

4. Identify loss scenarios (STPA Step 4)

UCA-1: Usuário não fornece fornecer credenciais quando o nome de usuário é fornecido, a senha é fornecida **[H-1]**

Cenário	*Fator causal*	*Recomendação*	*Justificativa*	*UCAs relacionadas*
A atualização automática do campo, falhou porque o usuário não percebeu que, mesmo após inserir o nome de usuário e senha,os dados não foram incorporados ao formulário.	Problema técnico na atualização automática da interface.	Realizar testes regulares para garantir que a interface seja atualizada automaticamente após a inserção inicial de credenciais, evitando confusão.	Assegura que a interface seja confiável, minimizando falhas técnicas que poderiam causar confusão ao usuário.	-
Interface confusa durante o registro, o usuário encontra dificuldades na interação com a interface de registro.	Problema de usabilidade na interface do registro.	Melhorar a interface, garantindo que os campos estejam claramente identificados e proporcionando uma experiência de usuário intuitiva.	Essa recomendação visa melhorar a usabilidade, facilitando para o usuário encontrar e inserir a senha durante o registro.	-
Problema técnico na validação do formulário, pois o usuário preenche corretamente o nome de usuário e a senha(control action emitida corretamente), mas por algum motivo não foi executado ou seguido adequadamente.	Há um problema técnico no processo de validação do formulário, que pode ser atribuído a vários fatores, incluindo um botão que não está funcionando corretamente, uma URL de destino inválida ou	Realizar testes regulares do sistema e implementar planos de contingência para resolver rapidamente problemas técnicos	Essa recomendação assegura que o sistema funcione corretamente, evitando frustrações do usuário causadas por falhas técnicas na validação do formulário.	-

	um problema no backend que resulta em Internal Server Error.			
Conta do usuário, está bloqueada temporariamente devido a tentativas de acesso malsucedidas anteriores. Isso impede o acesso, mesmo quando um nome de usuário e senha válidos são fornecidos.	Política de segurança que bloqueia contas após várias tentativas malsucedidas.	Implementar políticas de bloqueio de contas mais flexíveis e notificar os usuários sobre bloqueios para evitar confusão.	Melhora a segurança, mas mantém a transparência sobre bloqueios temporários de conta.	-

UCA-2: Usuário não fornece fornecer credenciais quando o nome de usuário não foi fornecido, a senha foi fornecida **[H-1]**

Cenário	***Fator causal***	***Recomendação***	***Justificativa***	***UCAs relacionadas***
Autenticação incompleta, pois o usuário não forneceu o nome de usuário, somente senha.	Falta de dados de entrada essenciais.	Exigir tanto o nome de usuário quanto a senha para concluir a autenticação.	Ambas as credenciais são necessárias para garantir a identificação segura do usuário.	UCA-3
Manipulação maliciosa, pois um ator tenta explorar uma possível vulnerabilidade fornecendo apenas a senha, sem incluir o nome de usuário.	Ação maliciosa na tentativa de contornar a autenticação.Como por exemplo, utilização da técnica de ataque SQL injection.	Reforçar medidas de segurança, como a implementação de verificações mais rigorosas e monitoramento de padrões suspeitos.	Protege contra tentativas maliciosas de manipulação no processo de autenticação.	**-**
Espaços em branco, o usuário ao tentar inserir nome de usuário houve espaços em branco.	Erro na validação de dados de entrada.	Implementar validação para rejeitar espaços em branco no nome de usuário.	Espaços em branco podem ser explorados, comprometendo a integridade da autenticação.	UCA-3
Força bruta na senha, pois um ator insere senha mas sem nome de usuário [H-1].	Tentativa sistemática de explorar a vulnerabilidade.	Exigir tanto o nome de usuário quanto a senha para concluir a autenticação.	A autenticação completa é vital para resistir a ataques de força bruta.	**-**

UCA-3: Usuário não fornece fornecer credenciais quando o nome de usuário é fornecido, a senha não é fornecida **[H-1]**

Cenário	***Fator causal***	***Recomendação***	***Justificativa***	***UCAs relacionadas***
Omissão intencional da senha, pois o usuário, de forma intencional, não fornece a senha.	Ação maliciosa do usuário para tentar acessar o sistema sem a senha.	Reforçar medidas de segurança, incluindo uma validação mais rigorosa dos dados de entrada e monitoramento de padrões suspeitos.	Protege contra tentativas intencionais de contornar a autenticação, garantindo que ambas as credenciais sejam fornecidas.	**-**
Desconsideração da segurança, pois a interface de autenticação falha em exigir a senha após nome de usuário.	Erro no processo de autenticação.	Reforçar a política de segurança, exigindo ambas as credenciais.	A segurança é comprometida quando apenas uma das credenciais é solicitada após o fornecimento do nome de usuário.	**-**
Erro de design na interface, pois a interface de autenticação não destaca adequadamente a necessidade de ambos os campos, levando o usuário a acreditar que pode autenticar-se fornecendo apenas o nome de usuário.	Design inadequado na interface	Revisar e otimizar a interface para garantir que os campos obrigatórios sejam claramente indicados e compreensíveis.	Melhora a usabilidade, evitando erros causados por problemas de design na interface de autenticação.	**-**

UCA- 4: Controlador de Login não fornece solicitar acesso quando o nome de usuário é fornecido, a senha é fornecida **[H-1]**

Cenário	***Fator causal***	***Recomendação***	***Justificativa***	***UCAs relacionadas***
Erro na comunicação com servidor, pois o controlador de login não consegue se comunicar efetivamente com o servidor, resultando na falta de acesso, mesmo com as credenciais corretas fornecidas.	Problemas de comunicação entre o controlador de login e o servidor.	Implementar monitoramento constante e alertas para identificar rapidamente problemas de comunicação e tomar medidas corretivas.	Garante uma resposta rápida a problemas técnicos que possam impedir o acesso, melhorando a disponibilidade do serviço.	**-**
Falha no processamento da autenticação,pois uma falha interna no processamento da autenticação impede que o controlador de login forneça acesso, mesmo com credenciais válidas.	Problema técnico no processo de autenticação.	Realizar auditorias regulares no sistema para identificar e corrigir problemas técnicos, garantindo o processamento adequado da autenticação.	Assegura que o sistema funcione corretamente, evitando falhas técnicas que possam comprometer a segurança.	**-**
Expiração inesperada da sessão, pois a sessão do usuário expira inesperadamente devido a configurações inadequadas, resultando na perda de acesso, mesmo com credenciais válidas.	Configuração inadequada da gestão de sessões.	Revisar e ajustar as configurações de gestão de sessões para evitar expirações prematuras e notificar os usuários sobre sessões prestes a expirar.	Melhora a experiência do usuário e evita a perda de acesso devido a expirações inesperadas da sessão.	**-**
Ataque de negação de serviço (DoS),pois um ataque de negação de serviço sobrecarrega o sistema, impedindo o controlador de login de fornecer acesso, mesmo com credenciais válidas.	Ataque externo que sobrecarrega os recursos do sistema	Implementar medidas de segurança, como firewalls e sistemas de detecção de intrusões, para mitigar e responder a ataques de negação de serviço	Protege contra ataques externos que poderiam comprometer a disponibilidade do sistema.	**-**

Problema de cache desatualizado, pois um cache desatualizado no controlador de login impede o reconhecimento de credenciais válidas, resultando na recusa de acesso.	Problema técnico no gerenciamento de cache.	Implementar estratégias de cache eficientes e automáticas que garantam a atualização regular para evitar problemas de cache desatualizado.	Melhora a eficiência do sistema, garantindo que as credenciais válidas sejam reconhecidas corretamente.	-

UCA- 5: Controlador de Login não fornece solicitar acesso quando o nome de usuário não é fornecido, a senha é fornecida **[H-1]**

Cenário	***Fator causal***	***Recomendação***	***Justificativa***	***UCAs relacionadas***
Problema de conectividade com o banco de dados, pois um problema de conectividade com o banco de dados impede o controlador de login de verificar o nome de usuário, resultando na recusa de acesso.	Problema técnico na infraestrutura do sistema.	Monitorar a conectividade do banco de dados e implementar redundâncias para garantir acesso contínuo, mesmo em caso de falhas temporárias.	Melhora a disponibilidade do sistema, minimizando interrupções causadas por problemas de conectividade.	**-**
Atualização do software desnecessária, pois uma atualização de software mal implementada afeta a lógica de verificação do nome de usuário, resultando na recusa de acesso.	Problema técnico na implementação de atualizações de software.	Realizar testes rigorosos antes de implementar atualizações e manter backups para reverter alterações em caso de problemas.	Garante que as atualizações de software não comprometam a capacidade do controlador de login de verificar o nome de usuário adequadamente.	**-**
Erro na lógica de autenticação,pois um erro de lógica no controlador de login impede a verificação adequada do nome de usuário, resultando na recusa de acesso.	Problema técnico na lógica de autenticação.	Realizar auditorias regulares no sistema para identificar e corrigir problemas técnicos na lógica de autenticação.	Assegura o funcionamento adequado do sistema, evitando falhas técnicas que possam comprometer a verificação do nome de usuário.	**-**

UCA- 6: Controlador de Login não fornece solicitar acesso quando nome de usuário não é fornecido, senha não é fornecida **[H-1]**

Cenário	***Fator causal***	***Recomendação***	***Justificativa***	***UCAs relacionadas***
Problema de configuração no front-end, pois uma configuração inadequada no front-end impede a correta transmissão das credenciais para o controlador de login.	Problema técnico na configuração da interface do usuário.	Realizar testes regulares na configuração do front-end para identificar e corrigir problemas técnicos que possam interferir na transmissão de credenciais.	Realizar testes regulares na configuração do front-end para identificar e corrigir problemas técnicos que possam interferir na transmissão de credenciais.	**-**
Ataque de injeção de dados pois um ataque de injeção de dados manipula as credenciais fornecidas pelo usuário, levando o controlador de login a recusar o acesso.	Ataque externo visando manipular as credenciais.	Implementar medidas de segurança, como filtragem de entrada, para prevenir ataques de injeção de dados.	Protege contra tentativas de manipulação de credenciais por parte de atacantes externos	**-**

UCA- 7: Controlador de Login não fornece solicitar acesso quando o nome de usuário é fornecido, a senha não é fornecida **[H-1]**

Cenário	***Fator causal***	***Recomendação***	***Justificativa***	***UCAs relacionadas***
Problema de conectividade com o banco de dados, pois um problema de conectividade com o banco de dados impede o controlador de login de verificar a senha, resultando na recusa de acesso.	Problema técnico na infraestrutura do sistema.	Monitorar a conectividade do banco de dados e implementar redundâncias para garantir acesso contínuo, mesmo em caso de falhas temporárias.	Melhora a disponibilidade do sistema, minimizando interrupções causadas por problemas de conectividade.	**-**
Atualização do software desnecessária, pois uma atualização de software mal implementada afeta a lógica de verificação da senha, resultando na recusa de acesso.	Problema técnico na implementação de atualizações de software.	Realizar testes rigorosos antes de implementar atualizações e manter backups para reverter alterações em caso de problemas.	Garante que as atualizações de software não comprometam a capacidade do controlador de login de verificar o nome de usuário adequadamente.	**-**
Erro na lógica de autenticação,pois um erro de lógica no controlador de login impede a verificação adequada da senha, resultando na recusa de acesso.	Problema técnico na lógica de autenticação.	Realizar auditorias regulares no sistema para identificar e corrigir problemas técnicos na lógica de autenticação.	Assegura o funcionamento adequado do sistema, evitando falhas técnicas que possam comprometer a verificação do nome de usuário.	**-**
Problema de cache desatualizado, pois um cache desatualizado no controlador de login impede o reconhecimento da senha, resultando na recusa de acesso.	Problema técnico no gerenciamento de cache.	Implementar estratégias de cache eficientes e automáticas que garantam a atualização regular para evitar problemas de cache desatualizado.	Melhora a eficiência do sistema, garantindo que o nome de usuário seja reconhecido corretamente.	**-**

UCA-8: Autenticador não fornece buscar usuário quando o nome de usuário e a senha são fornecidos. **[H4]**

Cenário	***Fator causal***	***Recomendação***	***Justificativa***	***UCAs relacionadas***
O Autenticador não consegue fornecer a ação de controle , pois o sistema está sofrendo ataques DDoS, fazendo com que o serviço se torne inacessível.	Sistema sofre ataques DDoS; Recursos do sistema limitados; Largura de banda insuficiente;	Distribuir tráfego em vários servidores; Monitoramento constante do tráfego da rede; Realizar testes regulares de resiliência e simulações de ataques DDoS; Limitar taxa de tráfego para conexões de entrada; Configurar firewall e IDS/IPS para bloquear tráfego malicioso.	Usar um balanceador de carga para distribuir o tráfego entre múltiplos servidores ajuda a evitar que um único servidor seja sobrecarregado durante um ataque DDoS; O monitoramento em tempo real pode ajudar a tomar medidas imediatas; Limitar taxas de tráfego pode ajudar a conter tráfego excessivo durante um ataque DDoS; Testes e simulações de ataque DDos ajudam a garantir que as medidas de segurança implementadas estejam corretas	UCA- 9; UCA-10; UCA-11
O Autenticador não fornece a ação de controle, pois não consegue se comunicar com o Repositório.	Erro no Repositório; Erros de leitura; Falha de comunicação entre componentes	A comunicação entre os componentes do sistema deve ser melhorada.	O sistema não pode ser mantido se não houver uma comunicação estável entre os componentes	UCA- 9; UCA-10; UCA-11

UCA-12: Autenticador fornece buscar usuário muito cedo quando o nome de usuário não é fornecido e a senha não é fornecido. **[H2]**

Cenário	***Fator causal***	***Recomendação***	***Justificativa***	***UCAs relacionadas***
As credenciais do usuário contém comandos SQL (SQLi) , fazendo com que um comando de busca seja executado, o que faz o Autenticador fornecer a ação de controle muito cedo.	O sistema sofre ataques de injeção SQL; Falta de validação de entrada; Falta de consultas parametrizadas; Erros de configuração e permissões; Mensagens de erro detalhadas;	Utilizar consultas parametrizadas ou preparadas; Validar e sanitizar dados de entrada; Evitar mensagens de erro detalhadas; Utilize firewalls de aplicação web (WAF); Monitore e registre atividades suspeitas: Teste de penetração.	Usar consultas parametrizadas separam os dados de entrada dos comandos SQL, tornando difícil para os atacantes injetar códigos; Validar e sanitizar os dados de entrada ajuda a garantir que os dados inseridos nas consultas SQL sejam seguros; Erros específicos do banco de dados podem revelar informações sensíveis que os atacantes podem usar para explorar vulnerabilidades; Os Web Application Firewalls podem ajudar a filtrar e bloquear solicitações maliciosas antes que atinjam sua aplicação, identificando padrões típicos de ataques de injeção de SQL; Testes regulares de penetração ajudam a identificar e corrigir vulnerabilidades de injeção de SQL.	UCA-17; UCA-18; UCA-19; UCA-20
O Autenticador fornece a ação de controle muito cedo, pois está sofrendo ataques de MITM, fazendo com que haja interceptação de requisições e	O sistema sofre ataques MITM; Comunicações não criptografadas; Uso de certificados	Criptografia de ponta a ponta; Usar autenticação de dois fatores (2FA); Monitorar tráfego de entrada	Usar criptografia de ponta a ponta em todas as comunicações faz com que as mensagens sejam criptografadas no ponto de	UCA-17; UCA-18; UCA-19; UCA-20

com isso, as ações de controle anteriores não são executadas.	digitais não confiáveis;	e saída por meio de Firewalls;	origem e descriptografadas apenas no ponto de destino, tornando difícil para um atacante interceptar ou modificar os dados; Autenticação de dois fatores adiciona uma camada extra de segurança, exigindo uma segunda forma de autenticação além das credenciais padrão; Uso de firewalls a impedir que o dispositivo seja invadido.	
O Autenticador recebe as credenciais com valores desatualizados, pois o Login Controller possui falhas.	Falha no Login Controller; Falha de comunicação entre componentes; Atrasos na entrega; Perda de mensagens; Sobrecarga de comunicação;	Implementar mecanismos de confirmação de entrega ou retransmissões automáticas; Projetar protocolos de comunicação eficientes e dimensionar a infraestrutura de rede e computação de maneira apropriada para minimizar latências;	Dados críticos podem ser perdidos devido a problemas de rede ou sobrecarga do sistema. Mecanismos de confirmação de entrega ou retransmissão automática podem garantir que os dados sejam entregues com sucesso; Projetar protocolos de comunicação eficientes e dimensionar a infraestrutura de rede e computação de maneira podem minimizar latências	-

UCA-13: Autenticador não fornece solicitar role quando o nome de usuário é válido e a senha é válida. **[H1]**

Cenário	***Fator causal***	***Recomendação***	***Justificativa***	***UCAs relacionadas***
O Autenticador possui erro no algoritmo, fazendo com que não haja verificação do role do usuário.	Algoritmo implementado incorretamente	Aplicação de testes para verificação da corretude do algoritmo; Verificação da lógica com stakeholders;	Aplicação de testes permite a validação do algoritmo	**-**
O Autenticador não fornece a ação de controle, pois está sofrendo ataques de MITM, fazendo com que a verificação de role não seja realizada devido à interceptação de requisições.	O sistema sofre ataques MITM; Comunicações não criptografadas; Uso de certificados digitais não confiáveis; Falha no algoritmo;	Criptografia de ponta a ponta; Usar autenticação de dois fatores (2FA); Monitorar tráfego de entrada e saída por meio de Firewalls;	Usar protocolos de criptografia para comunicações seguras garante que os dados transmitidos entre as partes sejam criptografados e não possam ser interceptados por um atacante.; Autenticação de dois fatores adiciona uma camada extra de segurança, exigindo uma segunda forma de autenticação além das credenciais padrão; Uso de firewalls para impedir que o dispositivo seja invadido.	**-**
O sistema sofre ataques DDoS, tornando-o indisponível e, consequentemente, fazendo com que o Autenticador não consiga fornecer a ação de controle para verificação de role.	Sistema sofre ataques DDoS; Recursos do sistema limitados; Largura de banda insuficiente;	Distribuir tráfego em vários servidores; Monitoramento constante do tráfego da rede; Realizar testes regulares de resiliência e simulações de ataques DDoS; Limitar taxa de tráfego para conexões de entrada; Configurar firewall e IDS/IPS	Usar um balanceador de carga para distribuir o tráfego entre múltiplos servidores ajuda a evitar que um único servidor seja sobrecarregado durante um ataque DDoS; O monitoramento em tempo real pode ajudar a tomar medidas imediatas; Limitar taxas de tráfego pode	**-**

		para bloquear tráfego malicioso.	ajudar a conter tráfego excessivo durante um ataque DDoS; Testes e simulações de ataque DDos ajudam a garantir que as medidas de segurança implementadas estejam corretas	

UCA-14: Autenticador fornece solicitar role quando o nome de usuário não é válido e a senha é válida. **[H2, H4]**

Cenário	***Fator causal***	***Recomendação***	***Justificativa***	***UCAs relacionadas***
O Autenticador possui erro no algoritmo, fazendo com que a ação de controle seja fornecida antes da verificação de credenciais do usuário.	Algoritmo implementado incorretamente	Aplicação de testes para verificação da corretude do algoritmo; Verificação da lógica com os *stakeholders;*	Aplicação de testes permite a validação do algoritmo;	UCA-15; UCA-16
O Autenticador sofre ataques de SQLi, na qual recebe variáveis (username password) contendo comandos SQL e executa a ação de controle antes de haver verificação das credenciais.	O sistema sofre ataques de injeção SQL; Falta de validação de entrada; Falta de consultas parametrizadas; Erros de configuração e permissões; Mensagens de erro detalhadas;	Utilizar consultas parametrizadas ou preparadas; Validar e sanitizar dados de entrada; Evitar mensagens de erro detalhadas; Utilize firewalls de aplicação web (WAF); Monitore e registre atividades suspeitas: Teste de penetração.	Usar consultas parametrizadas separam os dados de entrada dos comandos SQL, tornando difícil para os atacantes injetar códigos; Validar e sanitizar os dados de entrada ajuda a garantir que os dados inseridos nas consultas SQL sejam seguros; Erros específicos do banco de dados podem revelar informações sensíveis que os atacantes podem usar para explorar vulnerabilidades; Os Web Application Firewalls podem ajudar a filtrar e bloquear solicitações maliciosas antes que atinjam sua aplicação, identificando padrões típicos de ataques de injeção de SQL; Testes regulares de penetração ajudam a identificar e corrigir vulnerabilidades de injeção de SQL.	UCA-15; UCA-16

O Autenticador sofre ataques MITM, na qual intercepta as requisições de verificação de credenciais e faz com que a ação de controle *solicitar role* seja executada mesmo com credenciais não válidas.	O sistema sofre ataques MITM; Comunicações não criptografadas; Uso de certificados digitais não confiáveis	Criptografia de ponta a ponta; Usar autenticação de dois fatores (2FA); Monitorar tráfego de entrada e saída por meio de Firewalls;	Usar protocolos de criptografia para comunicações seguras garante que os dados transmitidos entre as partes sejam criptografados e não possam ser interceptados por um atacante.; Autenticação de dois fatores adiciona uma camada extra de segurança, exigindo uma segunda forma de autenticação além das credenciais padrão; Uso de firewalls para impedir que o dispositivo seja invadido.	UCA-15; UCA-16

UCA-21: Autorizador não fornece buscar role quando o nome de usuário é válido e senha é válida. **[H1, H6]**

Cenário	***Fator causal***	***Recomendação***	***Justificativa***	***UCAs relacionadas***
O Autorizador sofre ataques DDoS, fazendo com que a disponibilidade do seja comprometida e assim, não conseguindo prover sua ação de controle.	Sistema sofre ataques DDoS; Recursos do sistema limitados; Largura de banda insuficiente;	Distribuir tráfego em vários servidores; Monitoramento constante do tráfego da rede; Realizar testes regulares de resiliência e simulações de ataques DDoS; Limitar taxa de tráfego para conexões de entrada; Configurar firewall e IDS/IPS para bloquear tráfego malicioso.	Usar um balanceador de carga para distribuir o tráfego entre múltiplos servidores ajuda a evitar que um único servidor seja sobrecarregado durante um ataque DDoS; O monitoramento em tempo real pode ajudar a tomar medidas imediatas; Limitar taxas de tráfego pode ajudar a conter tráfego excessivo durante um ataque DDoS; Testes e simulações de ataque DDos ajudam a garantir que as medidas de segurança implementadas estejam corretas	-
O Autorizador sofre ataques MITM, fazendo com que a requisição realizada para emitir a ação de controle seja interceptada e assim, fazendo com que não seja emitida.	O sistema sofre ataques MITM; Comunicações não criptografadas; Uso de certificados digitais não confiáveis	Criptografia de ponta a ponta; Usar autenticação de dois fatores (2FA); Monitorar tráfego de entrada e saída por meio de Firewalls;	Usar protocolos de criptografia para comunicações seguras garante que os dados transmitidos entre as partes sejam criptografados e não possam ser interceptados por um atacante.; Autenticação de dois fatores adiciona uma camada extra de segurança, exigindo uma segunda forma de autenticação além das credenciais padrão; Uso de firewalls para impedir que	-

			o dispositivo seja invadido.	
O Autorizador sofre ataques de Força Bruta, fazendo com que o controlador não seja capaz de se manter após muitas requisições e assim, tendo sua disponibilidade comprometida e não conseguindo emitir sua ação de controle.	Autorizador sofre ataques de força bruta; Falta de políticas de senha;	Implementação de políticas de senhas robustas; Bloqueio de conta após tentativas sucessivas fracassadas; Autenticação de dois fatores (2FA); Captchas e/ou desafios de resposta humana; Limitação de tentativas por login;	Senhas mais robustas torna mais difícil encontrar a combinação correta por invasores; Bloquear conta após	-
O Autenticador recebe credenciais contendo códigos SQL, fazendo com que a verificação de role não seja feita e assim, o Autorizador não consegue emitir sua ação de controle.	O sistema sofre ataques de injeção SQL; Falta de validação de entrada; Falta de consultas parametrizadas; Erros de configuração e permissões; Mensagens de erro detalhadas;	Utilizar consultas parametrizadas ou preparadas; Validar e sanitizar dados de entrada; Evitar mensagens de erro detalhadas; Utilize firewalls de aplicação web (WAF); Monitore e registre atividades suspeitas: Teste de penetração.	Usar consultas parametrizadas separam os dados de entrada dos comandos SQL, tornando difícil para os atacantes injetar códigos; Validar e sanitizar os dados de entrada ajuda a garantir que os dados inseridos nas consultas SQL sejam seguros; Erros específicos do banco de dados podem revelar informações sensíveis que os atacantes podem usar para explorar vulnerabilidades; Os Web Application Firewalls podem ajudar a filtrar e bloquear solicitações maliciosas antes que atinjam sua aplicação, identificando padrões típicos de	-

			ataques de injeção de SQL; Testes regulares de penetração ajudam a identificar e corrigir vulnerabilidades de injeção de SQL.	
O Autorizador não consegue emitir sua ação de controle pois o Repositório está sofrendo com uma falha.	Erro no repositório; desastre natural; invasão no repositório; falha na comunicação entre componentes;	Melhoria na comunicação entre componentes; Utilização de backups; Melhoria na segurança do Repositório por meio de criptografia dos dados; Uso de conexões criptografadas para comunicação entre o Repositório e outros componentes;	Com o uso de backups, pode-se contornar a perda de dados caso o componente falhe; Uso de criptografia pode impedir ou dificultar o acesso indevido ao Repositório;	-

UCA-22: Autorizador fornece buscar role quando nome de usuário não é válido e senha é válida.**[H2, H3, H4, H6]**

Cenário	***Fator causal***	***Recomendação***	***Justificativa***	***UCAs relacionadas***
O Autenticador possui uma falha no algoritmo, permitindo a autenticação de um usuário não válido e assim, permitindo a verificação de *role* desse usuário pelo Autorizador.	Falha no algoritmo do autenticador;	Aplicação de testes para verificação da corretude do algoritmo;	Aplicação de testes permite a validação do algoritmo;	UCA-23; UCA-24
O Autorizador recebe credenciais com códigos SQL que fazem acesso direto ao Repositório, e assim, o Autorizador emite sua ação de controle.	Autorizador recebe códigos para SQLi; Falta de políticas de senhas robustas; Falha no algoritmo do Autenticador;	Melhoria do algoritmo do Autenticador para impedir a comunicação com o Autenticador quando houver credenciais não válidas; Implementação de políticas de senhas robustas;	O algoritmo de controle do Autenticador deve impedir o acesso ao Autorizador quando haver credenciais inválidas para impedir o acesso indevido; Políticas de senha robustas impede que as credenciais contenham códigos SQL;	UCA-23; UCA-24

UCA-25: Autorizador fornece buscar role muito cedo quando nome de usuário é fornecido e senha é fornecido. **[H4, H6]**

Cenário	***Fator causal***	***Recomendação***	***Justificativa***	***UCAs relacionadas***
Erro na comunicação entre Autenticador e o Autorizador. Esse erro resulta no Autorizador iniciar a busca de funções (roles) mesmo quando o processo de autenticação não é completado.	Falha na integração entre os componentes do sistema, o que está impossibilitando a transmissão adequada das credenciais para o autorizador.	Realizar uma revisão na integração entre os componentes do sistema para garantir a comunicação adequada das informações de autenticação.	Corrigir a comunicação entre os componentes é vital para garantir que o autorizador receba corretamente as credenciais antes de iniciar qualquer processo de autorização, evitando buscas de função prematuras.	UCA-26 UCA-27 UCA-28
Erro temporário no Autenticador e/ou no sistema pode fazer com que o Autorizador forneça a ação de controle. Isso decorre de uma falha lógica no algoritmo de controle do Autenticador, fazendo com que deixe de verificar as credenciais durante o período em que o Autenticador está inativo.	Falta de uma verificação específica impede a prevenção da busca de role quando a autenticação está temporariamente desativada	Reforçar a lógica de controle de acesso para considerar o estado da autenticação e evitar buscas de função durante períodos de desativação temporária.	Garantir que o sistema compreenda o status da autenticação evita autorizações inadequadas durante períodos em que a autenticação está desativada, protegendo contra acessos não autorizados.	UCA-26 UCA-27 UCA-28
O Autenticador excede o limite de tempo durante o processo de autenticação e não fornece sua ação de controle para o Autorizador. Com isso, o Autorizador,, conduz uma busca de funções antes de concluir completamente o processo de autenticação.	Ausência de uma gestão adequada do tempo durante o processo de autenticação.	Implementar um controle de tempo eficiente para garantir que o processo de autenticação não seja interrompido antes da conclusão.	Gerenciar corretamente o tempo durante a autenticação evita autorizações inadequadas e garante que a busca de função ocorra somente quando a autenticação for bem-sucedida.	UCA-26 UCA-27 UCA-28

Falha na lógica do algoritmo de controle do Autenticador, permitindo que o Autorizador forneça sua ação de controle mesmo quando a autenticação falha.	A robustez da lógica de controle de acesso é insuficiente para bloquear a busca de role quando a autenticação falha.	Reforçar a lógica de controle de acesso para garantir que a busca de funções só seja permitida após uma autenticação bem-sucedida.	Reforçar a lógica de controle de acesso garante que apenas usuários autenticados e autorizados possam buscar funções, aumentando a segurança do sistema.	UCA-26 UCA-27 UCA-28
Exposição a vulnerabilidades não descobertas. A busca de role antecipada pelo autorizador pode revelar vulnerabilidades desconhecidas no sistema antes que as correções adequadas sejam aplicadas, tornando o sistema suscetível a explorações maliciosas.	Este processo revela vulnerabilidades desconhecidas no sistema antes que correções adequadas sejam aplicadas, aumentando a suscetibilidade do sistema a possíveis explorações maliciosas.	Implementar uma abordagem de "bug bounty" ou programas de teste de penetração durante as fases iniciais para identificar e corrigir vulnerabilidades antes da autorização completa.	Permite uma resposta proativa a possíveis ameaças, melhorando a segurança global do sistema crítico.	UCA-26 UCA-27 UCA-28

Observação: Desejamos informar que, durante o processo de desenvolvimento do Passo 3, referente às UCA-13 e UCA-18 a UCA-21, foi dada atenção especial à estrutura delas. Embora estejamos cientes de que outras abordagens poderiam potencialmente aprimorar a redação das UCAs em questão, decidimos adotar a forma de escrita atual para assegurar a aderência ao método proposto no artigo "*A method based on Behavior Driven Development (BDD) and System-Theoretic Process Analysis (STPA) for verifying security requirements in critical software systems*".