Sistema de Auto-Reposição de ficheiros / Anti-Ransomware

Proposta de trabalho prático

Unidade curricular de Sistemas Distribuídos, prof. Pedro Rosa

Alunos: João Pires, 20200459

Diogo Massuça, 20200279

Hélio Divaldo, 20190928

Link de acesso ao repositório do projeto: https://github.com/joaoppiresp/ProjSistemas-Distr

Um dos grandes problemas atuais no mundo digital é a segurança, sendo as empresas alvo constante de ataques informáticos que infetam os seus sistemas e roubam dados importantes. Um dos mais utilizados métodos de ataque é o Ransomware. Ransomware toma várias formas, mas consistem todas nos mesmos passos base, isto é, penetrar um sistema sem autorização, infetá-lo com um virus que encripta todos os ficheiros possíveis e pedir um resgate aos donos dos ficheiros.

Este método é bastante eficaz pois muitas vezes as empresas não têm todos os seus sistemas com backups atualizados e redundâncias preparadas para este tipo de situação, nem têm garantia nenhuma que ao pagarem o resgate serão entregues as chaves de desencriptação nem que não voltaram a ser infetados.

Resumindo, o Ransomware é um ataque informático com a possibilidade de causar danos irreversíveis a um sistema.

Caso de uso

1 – Reposição de Ficheiros

Descrição	Caso de uso destinado a auto reposição de um ficheiro e passos inicias
Pré-condições	1. O utilizador possui uma conta na plataforma; 2. O utilizador tem acesso à internet;
Cenário Principal	1. Insira os dados do login e receba o código de autorização por email; 1.2 O utilizador é direcionado para a plataforma; 2. Caso o sistema verifica a identificação do utilizador, e a mesma corresponde a identificação que editou o ficheiro; 3. O sistema efetua um novo hash do ficheiro editado;
Cenário Alternativo	1.1. As credenciais do utilizador não se encontram registadas no sistema; A) O login dá erro 1.2. O utilizador introduz uma password incorreta; B) O login dá erro 1.3. O utilizador introduz o código de autorização incorreto;

	C) O login dá erro 1.4. As credenciais do utilizador não correspondem a identificação que editou o ficheiro registado no sistema; D) O sistema elimina o ficheiro editado e faz o backup do ficheiro;
Pós Condições	1. O utilizador tenta os passos novamente;
Cenários Excepcionais	- O utilizador sai da plataforma a qualquer momento;
|Pós Condições	- N/A;

SOLUÇÃO

Como solução a este problema propomos o desenvolvimento de um software que, como descrição base, funciona como um intermediário entre o acesso dos utilizadores de um sistema e o dito sistema e seus ficheiros.

Como primeira fronteira ao sistema, propomos que os utilizadores tenham que realizar login, sendo este processo previamente autorizado manualmente, com identificação de 2 fatores. As tentativas de acesso indevidas são bloqueadas pelo sistema e a informação do atacante recolhida para avaliação posterior.

Após um acesso autorizado, todas as alterações a ficheiros realizadas pelos utilizador serão registadas.

Os ficheiros terão backups automáticos, realizados periodicamente e atualizados consoante alterações feitas pelos utilizadores.

Para mitigar alterações não autorizadas, os hashs dos ficheiros serão guardados para comparação com os mais recentes. Se forem detetadas diferenças, o ficheiro alterado será apagado e reposto pelo original.

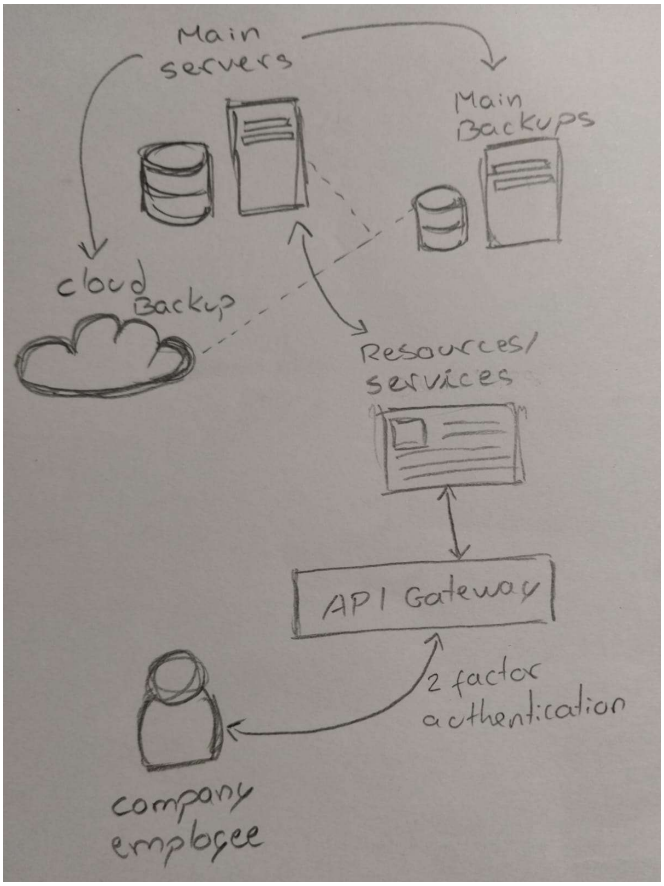
Os acessos ao sistema terão um time limit, sendo necessário o utilizador realizar novamente o login, passado algum tempo, para continuar a fazer alterações.

Todo este sistema enquadra-se bem com as unidades curricular lecionadas no presente semestre. Sistemas Distribuidos entra maioritariamente na necessidade do sistema ter redundâncias implementadas, como backups em vários lugares prontos a ser utilizados e necessidade de existir uma sincronização entre os ficheiros originais, alterados e em alteração. Sendo também de caráter bastante importante a segurança do sistema, como a utilização de autenticação 2 fatores por exemplo.

Requesitos Técnicos para desenvolvimento do Projeto

1. Autenticação 2 fatores (e-mail ou app, por decidir).
   Garante que o pedido de acesso ao sistema é feito por um utilizador legitimo.
2. Registo de utilizadores manual
   Deste modo é possível garantir que todos os utilizadores novos são realmente autorizados.
3. Backup de ficheiros
   Backups periodicos dos ficheiros de modo a salvaguardar a sua integridade no caso de alterações indevidas ou perda dos originais.
4. Controlo de tempo de vida de sessão
   Limitando o tempo de vida da sessão de um utilizador, obrigamos o utilizador a não manter ligação permanente ao sistema e a revalidar a sua presença ao longo do tempo.
5. Comparação de Hashs
   Ao armazenar um hash de cada ficheiro em atualização constante, é possível detetar alterações indevidas e seguir para eliminação dos ficheiros alterados e reposição da versão original.
6. Analise da integridade do sistema
   Através de uma análise periodica do sistema e seus ficheiros podemos garantir que o mesmo se mantém atualizado.
7. Obrigatoriedade de alteração de password
   Alterações periódicas de passwords de acesso dos utilizadores com confirmação de supervisor. Deste modo podemos garantir que apenas os utilizadores legítimos podem proceder à alteração das passwords por terem de ser confirmadas por supervisor. Ajudando também a manter as passwords sempre atualizadas.

Arquitetura da solução



Tecnologias a utilizar

Segue uma lista das tecnologias que pretendemos utilizar para o desenvolvimento do sistema:

8. Linux para os servidores;
   Versão ainda por decidir.
9. AWS para cloud backups;
   Utilizado para demonstração do sistema em escala pequena.
10. SpringBoot para desenvolver a REST API;
    Ligação servers-utilizador-
11. HAPROXY para segurança;
12. Software de autenticação 2 fatores;

Link de acesso ao Planeamento e Calendarização, incluindo distribuição de tarefas no Excel:

https://iade-my.sharepoint.com/:x:/r/personal/20190928_iade_pt/_layouts/15/Doc.aspx?sourcedoc=%7B80a5d007-a79b-41f6-b473-b54d0fc2f1ed%7D&action=edit&wdPreviousSession=b2e2a1a1-0fa3-4e9a-92ce-3e685352a08e