Licenciatura em Tecnologias Digitais e Segurança de Informação

Turma do 2º Ano, 2º Semestre 2024/25

Trabalho realizado no âmbito das Unidades Curriculares:

- Sistemas Distribuídos e Segurança, com Professor/Investigador João Pedro Pavia
- Programação para a Internet, com Professor/Doutor Thiago Bessa Pontes
- Criptografia Aplicada, com Professor/Autor/Doutor Carlos Serrão

Trabalho realizado por:

- Eliaquim Alexandre, Nº 122051
- Gonçalo de Monção Meireles Liberato Ferreira, Nº 120037

GRUPO 8

Tema do trabalho:

Sistema de Armazenamento Seguro de Informação na Cloud

Índice

Introdução

Para desenvolver um sistema de armazenamento seguro de informação na Cloud, com a finalidade de o vender como um serviço, o grupo constou que deve ter algumas considerações chave em conta.

Este trabalho tem como objetivo o desenvolvimento de um **Sistema de Armazenamento Seguro de Informação na Cloud**, visando oferecer um serviço que garanta alta disponibilidade, proteção de dados e conformidade com regulamentações como o **RGPD**. Para isso, foram analisadas soluções existentes, identificadas lacunas no mercado e projetada uma arquitetura robusta, baseada em tecnologias avançadas como **Kubernetes**, VPN híbrida e criptografia de ponta a ponta.

As maiores preocupações que devem ser adereçadas, na procura do desenvolvimento de uma plataforma deste tipo, são:

- A escalabilidade da infraestrutura;
- A segurança e conformidade dos clientes com as regras impostas;
- Confiabilidade e disponibilidade do sistema;
- Arquitetura e design da rede para performance excelente;
- Custo de gestão e monitorizamento;
- Suporte e customer service;
- API´s e integração;
- Armazenamento de dados e backup dos mesmos;
- Diferenciabilidade do mercado, Interface Amigável;

Ao longo deste relatório, serão discutidos os principais desafios do projeto, as estratégias adotadas para mitigar riscos e a implementação de um modelo eficiente que combina desempenho, segurança e usabilidade. Além disso, serão apresentadas as medidas adotadas para garantir a escalabilidade do sistema, a proteção contra ataques cibernéticos e a diferenciação no mercado através de uma interface intuitiva e um serviço de apoio eficiente.

DESCRIÇÃO DO PROBLEMA E PROPOSTA DE SOLUÇÃO

ESCALABILIDADE

Um serviço Cloud deve ser capaz de suportar um aumento substancial de carga sem que o seu desempenho piore ao ponto de pôr em causa a sua utilização. Se fosse possível, o grupo queria procurar escalabilidade também de um ponto de vista económico, ou seja, priorizar replicação de processos com pouca ou nenhuma necessidade de novos investimentos, uma vez que investimento será crucial noutras áreas ao longo do processo de desenvolvimento do serviço.

Para isto podem ser consideradas diversas "Cloud-Native Infrastructures", um termo que se refere a hardware e software que suporta aplicações concebidas especificamente para ambientes de Cloud, permitindo que estas possam beneficiar das vantagens oferecidas por modelos de "Cloud computing" e se destaquem especialmente na consideração chave que é a escalabilidade.

No entanto, ao descobrir acerca das valiosas funcionalidades que oferece o sistema open source "Kubernetes", o grupo percebeu imediatamente que esta seria a escolha acertada.

Este sistema agrupa containers para construção de aplicações, através da divisão de tarefas por unidades lógicas em cada container, assim permitindo escalabilidade fácil nos dois sentidos e ainda uma gestão mais simples. Isto acontece graças a uma característica designada por “auto-scaling”, que consiste na criação automática de mais containers/pods à rede cada vez que, para esta funcionar, se comecem a utilizar mais recursos do servidor do que os previstos pela equipa. Por exemplo, se configurarmos a rede para utilizar 50% da capacidade de processamento de cada container para servidores de Back-End e este necessitar de mais para cumprir com os limites de tempo propostos, automaticamente é criado e adicionado à rede um novo container, com o qual os restantes vão dividir tarefas.

Para além disso permite escolher e utilizar diversos sistemas de armazenamento simultaneamente, não obstante da localização dos mesmos. Este era um objetivo desde o início.

SEGURANÇA E CONFORMIDADE COM AS REGRAS IMPOSTAS (GDPR)

Por norma, um serviço de Cloud-providing concebe e/ou dispõe diversas categorias de servidores a um cliente interessado na sua utilização. Se este considera que uma subscrição ao serviço é uma mais valia, de certo é porque espera confidencialidade, disponibilidade e integridade dos seus dados, possivelmente sensíveis e de grande valor. Trata-se de uma tarefa de extrema importância, que envolve planeamento e gestão de risco, bem como investimento em componentes soft e hardware.

Entre outros, mas talvez o mais relevante do ponto de vista de um cliente, o servidor de DB (Data Base, Base de Dados) é disponibilizado pelos CSPs com a finalidade de armazenar as informações do proprietário num local seguro, para que este disponha de um maior espaço para alocação de ficheiros.

Quando se procura mitigar ameaças relativas a vazamento de dados e violações de privacidade, existem algumas soluções já existentes pelas quais podemos optar. Pensou-se que medidas de segurança robustas seriam as mais adequadas para o nosso projeto- que tipo de criptografia e quando, onde colocar firewalls e/ou honeypots, se é pertinente a existência de um log-management server, e, já que a maioria dos routers suporta a tecnologia VPN, se deveria ser utilizada.

Todo o processo de funcionamento do sistema de armazenamento de informação em Cloud do nosso grupo está explicado na secção de “Arquitetura e Design da Rede”. Aqui estão fundamentadas as nossas escolhas:

O primeiro passo é assegurar que apenas o proprietário de determinada conta consegue aceder aos dados que o mesmo tem guardados nas bases de dados. Existem várias formas de alcançar este objetivo, sendo algumas delas:

Tokens físicos ou JWTs (JSON Web Tokens), autenticação biométrica, OTP (One Time Password), etc.

Examinou-se o conjunto de possíveis opções e chegou-se à conclusão de que troca de chaves publicas e privadas, com validação perante uma Autoridade de Certificados (CA) para autenticar as extremidades da VPN (Gateway local e Gateway da Cloud), seria a melhor forma de o fazer.

1. A criptografia assimétrica permite a criação de assinaturas digitais que podem ser utilizadas para verificar a autenticidade e a integridade de uma mensagem. Isto garante que a mensagem foi enviada pelo remetente alegado e que não foi alterada durante a transmissão.

2. Assinaturas digitais também garantem integridade das mensagens.
3. Como apenas o detentor da chave privada pode criar uma assinatura digital válida, o remetente não pode negar ter enviado a mensagem (não-repúdio).
4. Em sistemas com muitos utilizadores, a criptografia assimétrica simplifica a gestão de chaves. Cada utilizador tem um par de chaves (pública e privada), e a chave pública pode ser distribuída livremente, enquanto a chave privada é mantida em segredo.
5. A criptografia assimétrica é compatível com uma variedade de sistemas e plataformas, facilitando a interoperabilidade entre diferentes tecnologias.
6. Como a chave pública pode ser partilhada sem receio, a criptografia assimétrica é ideal para comunicações seguras em canais públicos ou inseguros, como a internet. Para o caso do projeto que a equipa se encontra a desenvolver é o ideal, uma vez que este método será utilizado para autenticação do utilizador antes de o protocolo IPSec ter oportunidade de adicionar uma camada de segurança.

O sistema vai implementar K8s (Kubernetes) que comunicam entre si através do protocolo NodePort, ou seja, cada nó tem um IP (associado ao container em que a app está a correr) e uma porta (associada à aplicação específica que se quer adereçar dentro desse container, pois estarão a correr muitas aplicações simultaneamente). Para tornar segura a comunicação entre nós será implementado o protocolo IPSec em cada um com as configurações que forem necessárias para uma troca de informações através de conexão dedicada. Caso se verifique que o Service (balanceador de carga interno) tem dificuldade a comunicar rapidamente com os vários pods, trocar-se-á para uma comunicação VPN Over-The-Internet.

Modo de utilização da VPN:

Uma escolha popular é a VPN tradicional, uma vez que se trata de uma solução económica. Constatou-se, no entanto, que esta pode ter limitações em termos de desempenho e confiabilidade devido à dependência da internet pública. Outra opção comum é a conexão dedicada. Esta destaca-se uma vez que oferece uma ligação privada e de alta performance entre a infraestrutura local e a nuvem. Raciocinou-se a utilização da mesma, sendo que ferramentas de implementação desta tecnologia asseguram maior nível de segurança, porém ao observar que a sua implementação pode ser extremamente complexa e cara para um sistema de armazenamento Cloud em rápida expansão (má escalabilidade), optámos por uma mistura dos dois métodos.

Uma Hybrid VPN Connection é uma solução de rede que combina uma rede privada virtual (VPN) com uma conexão dedicada, como uma linha privada ou uma conexão de rede de longa distância (WAN), para criar uma infraestrutura de rede segura e flexível. **Uma conexão dedicada para tráfego crítico, uma conexão VPN Over-the-Internet para substituição no caso de falha ou para tráfego não crítico.**

Está fundamentada na secção da segurança a nossa preferência pelo protocolo IPSec, com IKE 2.0 para derivação da chave simétrica, bem como a utilização de autoridades de certificado (CA, Certificate Authorities) para validação de chaves assimétricas publicas. Como tal, o objetivo do grupo é uma arquitetura de rede que permita aos utilizadores autenticarem-se perante a Hybrid-VPN mediante estas entidades e de acordo com estes protocolos antes de terem acesso aos servidores Front End

CONFIABILIDADE E DISPONIBILIDADE DO SISTEMA

Deve ser implementado um plano de ação que evite downtimes, e ainda backup servers com a finalidade de agir como um anti falhas. Pretende-se criar redundância e mecanismos de fácil escalabilidade na preparação da arquitetura do sistema, e para isso decidiu-se utilizar uma API open source chamada Kubernetes.

Esta tecnologia é útil uma vez que, para além da auto-scaling, tem a propriedade de ser “self-healing”. Ou seja, se num conjunto de três containers apenas um deles estiver a correr um servidor de Front End e o mesmo falhar, um dos restantes containers vai desempenhar a função do primeiro até que o problema esteja resolvido. Este conjunto divide as tarefas de forma inteligente de forma a reduzir ao máximo a carga de trabalho de cada container individual, através de um control-loop que será desenvolvido pelo grupo.

ARQUITETURA E DESIGN DA REDE

Trata-se de um sistema distribuído, no sentido em que é um sistema assíncrono e heterogéneo, existe concorrência (vários clientes podem usar o sistema simultaneamente) e partilha de recursos (hardware, software, dados), e as falhas são (idealmente) independentes.

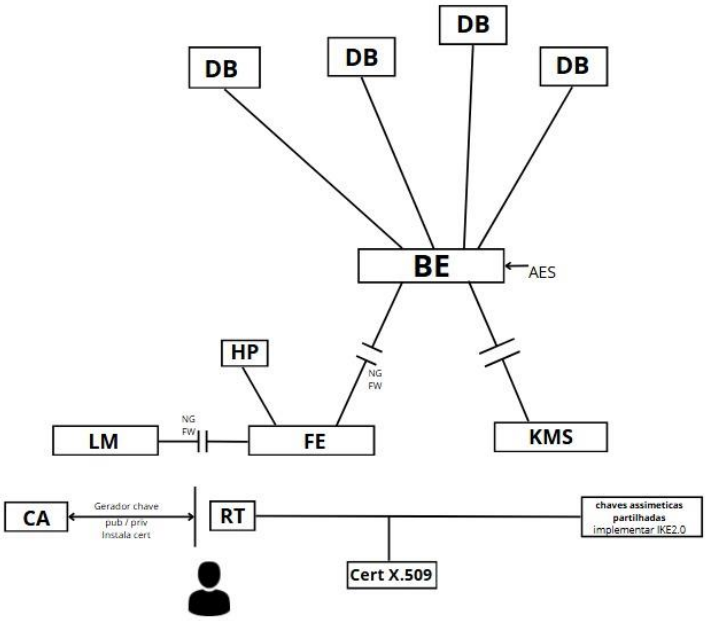
O modelo de programação é Cliente/Servidor, e a comunicação entre componentes é feita através de mensagens (FTP).

Componentes Necessários:

- Gateway com suporte VPN no ambiente local e no ambiente de Cloud;
- Certificados Digitais emitidos por uma CA (DigiCert);
- Kubectl para configurar nós de K8;
- Biblioteca OpenSSL para configuração dos algoritmos de criptografia;
- Em dispositivos Linux, como no nosso caso, OpenSwan p/ configuração do protocolo IPSec;
- Software OpenVPN, utilizado na configuração da VPN, implementação do IPSec, método de autenticação com certificados;
- Hardware para oferecer recursos a IDS/IPS ao Honeypot/Firewall/Log Management Server;
- pfSense, contém interfaces amigáveis para implementação do IPSec e permite modificação;
- Kippo HoneyPot e as suas dependências (Python 3.11 e OpenSSL)
- Ambiente de testes, onde será utilizado Wireshark, ping/traceroute, Log testing;

Clientes esperam uma latência baixa e velocidades de transferência elevadas. Uma performance de rede fraca leva a uma experiência frustrante e possivelmente à perda de clientes.

Porém, como já foi referido, a segurança é um pilar que deve estar presente em todos os processos de comunicação do sistema de armazenamento. Assim sendo, CSPs procuram diversas maneiras de implementar o melhor dos dois mundos nos seus designs.



1.0 Representação topológica do sistema distribuído

- O utilizador vai autenticar-se perante uma Autoridade de Certificação (CA) para validar a sua chave pública. Recebe um certificado e guarda-o.
- Após esta autenticação, deriva uma chave simétrica através da negociação de chaves com o servidor de Back End (BE), que utilizará para encriptar o certificado e autentica-se usando o protocolo IKEv2.
- O login é feito pela interface desenvolvida no servidor Front End (FE), onde o utilizador insere as credenciais. O FE age como um proxy seguro, estando estabelecida uma Firewall entre este e o BE para evitar a passagem de payloads maliciosos na transferência destas credenciais para o servidor onde serão processadas.
- É então estabelecida uma VPN híbrida entre o Gateway do cliente e o Gateway da Cloud, utilizando o modo túnel do IPSec. Periodicamente a chave simétrica será alterada, conforme configurado no protocolo IKE.
- Um servidor de Log management (LM) está associado diretamente ao servidor de FE, para controlo de sessões, defendido por uma firewall.
- Um HoneyPot com portas abertas (p. E. port 2222), uma interface de administração e uma réplica da tabela de logs do LM está também associado ao servidor FE, indefeso.

Já tendo OpenSSL instalado, configurar-se-á um Kippo HoneyPot para esta finalidade.

- O Key Management Server (KMS) usa um esquema de criptografia de envelope para encriptar dados em etcd. Depois de recebidos, os ficheiros são encriptados usando uma chave de criptografia de dados (DEK).
- Uma vez que existe uma DEK para cada ficheiro, os DEKs são encriptados de seguida com uma chave de criptografia de chaves (KEK) que é armazenada e gerida com as restantes no KMS, este protegido por FW.
- Por fim, ficheiros já encriptados são fragmentados e distribuídos por diversas DBs, sendo-lhes atribuído um número de processo e sequência. Backups feitos regularmente.

CUSTO DE GESTÃO E MONITORIZAMENTO

Embora esta parte não influencie diretamente os clientes, continua a ser essencial fazer a gestão dos recursos da rede e dos investidores.

Sabe-se que o CSP deve ser capaz de assegurar disponibilidade, integridade e confidencialidade, uma boa monitorização permite uma garantia de que estamos em conformidade com a nossa parte do acordo e conseguimos agir pronta e antecipadamente.

Tanto a implementação do sistema de Kubernetes como o Log-Management Server são essenciais para esta consideração.

SUPORTE E CUSTOMER SERVICE

Clientes podem necessitar de ajuda a habituar-se à interface, com troubleshooting, ou até com a documentação do GDPR. É difícil prever que problema o próximo cliente vai ter, mas disponibilizar uma ajuda rápida e eficiente, seja através de uma equipa preparada para atendimento ao telefone, live chat ou auxílio com Inteligência Artificial é muito importante.

Uma vez que a equipa desenvolvedora consiste em apenas dois elementos, esta considerou programar um bot que ajude o cliente com qualquer dúvida que possa ter. Caso a ajuda não seja suficiente, tanto o contacto de telefone como e-mail estarão presentes no final da página e na secção de “suporte e serviço ao cliente”.

API´s E INTEGRAÇÃO

API significa "Aplication Programming Interface". Quando duas aplicações precisam de comunicar uma com a outra, para garantir o bom funcionamento de um serviço, utilizam-se as API´s. Consistem num conjunto de protocolos e algoritmos que estruturam um sistema de comunicação entre duas apps, para que estas sabam lidar com requests e fazer responses adequadas.

Algumas soluções populares para este problema são ferramentas de desenvolvimento de easy-to-use API keys, para provisionar, monitorizar e escalar os recursos da cloud automáticamente.

Eis alguns exemplos: Google Analytics API e Mixpanel (para análise de eventos e comportamentos do utilizador), IBM Watson API (para análise de texto, reconhecimento de fala, e assistentes virtuais), Auth0 (sara autenticação e autorização, incluindo login social, single sign-on (SSO), e gestão de utilizadores). Serão desenvolvidas APIs pela própria equipa desenvolvedora do projeto.

Armazenamento de dados e backup dos mesmos

A perda de dados é um desafio sério para qualquer empresa, podendo resultar de falhas de hardware, ataques cibernéticos ou erro humano. Sem um sistema de backup eficaz, há risco de interrupção das operações, perda de informações valiosas e danos à reputação da organização.

Uma solução recomendada para o nosso projeto é o **Veeam Backup & Replication**. Esta aplicação permite realizar backups automáticos, incrementais e seguros, protegendo dados em ambientes físicos, virtuais e em nuvem. Além disso, oferece recuperação rápida em caso de falhas, garantindo a continuidade do negócio.

Também assegura a conformidade com normas de proteção de dados, através de funcionalidades de encriptação e controlo de acesso, tornando-se uma escolha robusta e confiável para a nossa empresa.

A nossa empresa irá implementar o Veeam Backup & Replication como solução principal, realizando cópias automáticas diárias e armazenando-as tanto localmente como na cloud para garantir segurança e redundância. Serão definidas políticas claras de backup e serão feitos testes periódicos para assegurar que o serviço não é interrompido.

Diferenciabilidade do mercado, Interface Amigável

Num mercado competitivo, a diferenciabilidade é fundamental para destacar uma empresa face à concorrência. Oferecer um produto ou serviço com uma interface amigável aumenta a satisfação do utilizador, reduz a curva de aprendizagem e melhora a experiência no geral. Sem uma interface bem concebida, o utilizador pode sentir-se confuso ou desmotivado a utilizar a solução, o que afeta diretamente a retenção e o sucesso do negócio.

Figma é uma ferramenta de design UI/UX muito popular, porém só está disponível para Windows e macOS. Adobe XD e Sketch são outras com grande reconhecimento no mercado, porém, após consideração, a equipa não achou pertinente fazer o download ou utilizar uma página web de uma ferramenta para desenvolver uma User Interface.

Tendo Visual Studio Code previamente instalado.

Durante o desenvolvimento do sistema procurar-se-á um design amigável, limpo, organizado e interativo para a interface, construída com HTML e CSS. A utilização destes recursos possibilita uma personalização visual que reforça a identidade da marca, contribuindo para a diferenciabilidade no mercado.

REFERÊNCIAS

[1] Allen, T. A. (2010, November 15). *NIST Cloud Computing Program - NCCP*. NIST. https://www.nist.gov/programs-projects/nist-cloud-computing-program-nccp

[2] National Institute of Standards and Technology (NIST). (2024). *Cybersecurity Framework*. National Institute of Standards and Technology. https://www.nist.gov/cyberframework

[3] Cloud Security Alliance. (n.d.). *CSA Security Guidance*. Cloud Security Alliance. https://cloudsecurityalliance.org/research/guidance

[4] *Top Threats | Cloud Security Alliance*. (2016). Cloud Security Alliance. https://cloudsecurityalliance.org/research/working-groups/top-threats

[5] *Decision Making Tools for Mission Critical Priorities*. (n.d.). Gartner. https://www.gartner.com/en/tools

[6] *Interoperable EU Risk Management Toolbox*. (n.d.). ENISA. https://www.enisa.europa.eu/publications/interoperable-eu-risk-management-toolbox

[7] Amazon Web Services. (2023). *AWS Well-Architected - Build secure, efficient cloud applications*. Amazon Web Services, Inc. https://aws.amazon.com/architecture/well-architected/?wa-lens-whitepapers.sort-by=item.additionalFields.sortDate&wa-lens-whitepapers.sort-order=desc&wa-guidance-whitepapers.sort-by=item.additionalFields.sortDate&wa-guidance-whitepapers.sort-order=desc

[8] Amazon Web Services. (2019). *Cloud Security – Amazon Web Services (AWS)*. Amazon Web Services, Inc. https://aws.amazon.com/security/

[9] *Cloud Security Best Practices Center*. (n.d.). Google Cloud. https://cloud.google.com/security/best-practices

[10] *Overview | Architecture Framework*. (n.d.). Google Cloud. https://cloud.google.com/architecture/framework

[11] TerryLanfear. (n.d.). *Azure security documentation*. Learn.microsoft.com. https://learn.microsoft.com/en-us/azure/security/

[12] bennage. (n.d.). *Azure Architecture Center - Azure Architecture Center*. Learn.microsoft.com. https://learn.microsoft.com/en-us/azure/architecture/

[13] European Union. (2018). *General Data Protection Regulation (GDPR)*. General Data Protection Regulation (GDPR). https://gdpr-info.eu/

[14] AWS. (2024). *What Is AWS Key Management Service? - AWS Key Management Service*. Docs.aws.amazon.com. https://docs.aws.amazon.com/kms/latest/developerguide/overview.html

[15] for. (2024, September 13). *Using a KMS provider for data encryption*. Kubernetes. https://kubernetes.io/docs/tasks/administer-cluster/kms-provider/#configuring-the-kms-provider-kms-v2

[16] and, S. (2024). *Cryptographic Module Validation Program | CSRC*. Nist.gov. https://csrc.nist.gov/projects/cryptographic-module-validation-program/certificate/4884

[17] *Sophos | Fully Synchronized, Cloud-Native Data Security*. (n.d.). SOPHOS. https://www.sophos.com/en-us