Documentação detalhada da arquitetura de Segurança na AWS

Essa documentação inclui as políticas, procedimentos e a integração entre os serviços, além de descrever o propósito de cada um.

Visão Geral:

Esta arquitetura foi desenhada para garantir a alta disponibilidade, segurança e conformidade para uma aplicação de e-commerce da startup Nova Tech hospedada na AWS. O foco principal é a proteção de dados sensíveis, otimização de custos e a implementação de políticas de segurança robustas para prevenir acessos não autorizados e mitigar ameaças cibernéticas.

Componentes Principais:

1. **Amazon Route 53**:
    - **Função**: Gerenciamento de DNS.
    - **Integração**: Está diretamente ligado ao CloudFront para o roteamento do tráfego da web para o domínio da aplicação.
    - **Políticas**: Configuração de failover para garantir a alta disponibilidade, direcionando o tráfego para regiões saudáveis em caso de falhas.
2. **Amazon CloudFront**:
    - **Função**: Rede de distribuição de conteúdo (CDN) para cache de conteúdo estático.

- **Integração**: Ligado ao S3 para a entrega de arquivos estáticos e ao Web Application Firewall (WAF) para proteção contra ataques na camada de aplicação.
- **Políticas**: Regras de WAF para bloquear tráfego malicioso e integração com TLS/SSL para tráfego criptografado.

3. **Amazon S3**:
   - **Função**: Armazenamento de objetos estáticos, como imagens e vídeos.
   - **Integração**: Servido pelo CloudFront para distribuição de conteúdo, e seus dados são criptografados com AWS Key Management Service (KMS).
   - **Políticas**: Aplicação de políticas de bucket para garantir que apenas usuários autorizados tenham acesso aos dados e habilitação de versionamento para recuperação de dados.
4. **AWS WAF**:
   - **Função**: Firewall de Aplicação Web.
   - **Integração**: Ligado ao CloudFront para bloquear tentativas de ataque na camada de aplicação, como SQL Injection e Cross-Site Scripting (XSS).
   - **Políticas**: Criação de regras personalizadas para bloquear IPs maliciosos, bloquear tráfego suspeito e definir limites de taxa de requisições.
5. **VPC (Virtual Private Cloud)**:
   - **Função**: Isolamento de rede para os recursos da AWS.
   - **Integração**: Contém subnets públicas e privadas, além de gateways NAT e Internet para controle de tráfego.
   - **Políticas**: Regras de segurança com grupos de segurança e ACLs (Access Control Lists) para controlar o tráfego de entrada e saída, garantindo que apenas tráfego autorizado atinja os recursos.
6. **Internet Gateway**:
   - **Função**: Permitir que instâncias em subnets públicas acessem a internet.
   - **Integração**: Conectado às subnets públicas que hospedam os balanceadores de carga.

    - **Políticas**: Controlar o acesso por meio de grupos de segurança e ACLs da VPC.
7. **NAT Gateway**:
    - **Função**: Permitir que instâncias em subnets privadas acessem a internet sem se tornarem acessíveis diretamente da internet.
    - **Integração**: Conectado às subnets privadas onde os servidores EC2 e RDS estão localizados.
    - **Políticas**: Regras de segurança aplicadas nas subnets privadas para controlar o tráfego de saída.
8. **Elastic Load Balancer (ELB)**:
    - **Função**: Balanceamento de carga de tráfego para distribuir as requisições entre várias instâncias de EC2.
    - **Integração**: Conectado ao CloudFront para distribuir tráfego de entrada e conectado às instâncias EC2 nas subnets privadas.
    - **Políticas**: Configuração de HTTPS com certificados TLS/SSL para tráfego seguro, e integração com CloudWatch para monitoramento de métricas de performance.
9. **Amazon EC2**:
    - **Função**: Instâncias de computação para rodar a aplicação de e-commerce.
    - **Integração**: Conectado ao RDS e ao Elastic Load Balancer para hospedar a aplicação. Está dentro das subnets privadas da VPC para aumentar a segurança.
    - **Políticas**: Aplicação de políticas de segurança através de grupos de segurança restritivos e criptografia de volumes com EBS.
10. **Amazon RDS (Relational Database Service)**:

- **Função**: Banco de dados relacional (usando MySQL, PostgreSQL ou Aurora).
- **Integração**: Conectado às instâncias EC2 para armazenar dados da aplicação, com replicação multi-AZ para alta disponibilidade.
- **Políticas**: Criptografia em repouso com KMS e políticas de backup automáticas para garantir a recuperação de dados.

11. **Amazon DynamoDB**:

- **Função**: Banco de dados NoSQL.
- **Integração**: Usado para armazenar dados não relacionais como sessões de usuários e dados de cache.
- **Políticas**: Controle de acesso baseado em identidades com AWS IAM e criptografia automática de dados com KMS.

12. **AWS Systems Manager**:

- **Função**: Gerenciamento de instâncias EC2 e automação de tarefas operacionais.
- **Integração**: Conectado às instâncias EC2 e Lambda para automação de respostas a incidentes.
- **Políticas**: Políticas de acesso restrito com IAM e automação de correção de falhas.

13. **AWS Lambda**:

- **Função**: Execução de código sem servidor em resposta a eventos.
- **Integração**: Integrado ao Systems Manager e a outras partes do sistema para executar automações baseadas em eventos.
- **Políticas**: Definição de permissões através do IAM para garantir que só execute as funções autorizadas.

14. **Amazon CloudWatch**:

- **Função**: Monitoramento e geração de logs para a infraestrutura.
- **Integração**: Conectado a todos os serviços da arquitetura, incluindo EC2, ELB e RDS, para monitoramento de métricas e alarmes.
- **Políticas**: Configuração de alarmes para alertar a equipe em caso de falhas ou anomalias.

15. **AWS CloudTrail**:

- **Função**: Auditoria e monitoramento de atividades da conta AWS.
- **Integração**: Monitora todas as ações realizadas nos serviços da AWS, fornecendo trilhas de auditoria.
- **Políticas**: Habilitado em toda a conta AWS para garantir que todas as ações sejam registradas e auditáveis.

16. **AWS Config**:

- **Função**: Monitoramento da conformidade dos recursos da AWS com as políticas definidas.
- **Integração**: Conectado a todos os serviços para garantir conformidade contínua com políticas de segurança e auditoria.
- **Políticas**: Regras configuradas para verificar se os recursos seguem as melhores práticas e políticas corporativas.

17. **AWS Key Management Service (KMS)**:

- **Função**: Gerenciamento de chaves criptográficas.
- **Integração**: Usado para criptografar dados no S3, EBS, RDS e DynamoDB.
- **Políticas**: Políticas de acesso para garantir que apenas usuários e serviços autorizados possam acessar as chaves de criptografia.

Fluxo e Conectividade:

1. **Usuário** acessa o site através do **CloudFront** (CDN), que distribui o tráfego para o **Elastic Load Balancer (ELB)**.
2. O ELB balanceia o tráfego entre instâncias **EC2**, que hospedam a aplicação dentro das subnets privadas na **VPC**.
3. As instâncias EC2 se conectam ao banco de dados relacional **Amazon RDS** (primário e secundário) ou ao banco **DynamoDB** para buscar ou armazenar dados.
4. **Amazon CloudFront** usa **AWS WAF** para bloquear tráfego malicioso e proteger a aplicação contra ameaças da web.
5. **CloudWatch** e **CloudTrail** monitoram a atividade e o desempenho de toda a infraestrutura, alertando sobre qualquer atividade suspeita.
6. **AWS Systems Manager** e **Lambda** automatizam respostas a incidentes e gerenciam a operação das instâncias EC2.
7. **S3** armazena conteúdo estático, com criptografia gerenciada pelo **KMS**.

Procedimentos de Segurança:

- **Autenticação e Autorização**: Implementação de **IAM Roles** para controlar o acesso aos serviços AWS com o princípio do menor privilégio.
- **Monitoramento Contínuo**: Uso de **CloudTrail** e **CloudWatch** para auditoria e monitoramento de atividades em tempo real.
- **Resposta a Incidentes**: Integração de **Systems Manager** com **Lambda** para respostas automáticas a falhas e incidentes de segurança.
- **Criptografia**: Implementação de criptografia de dados em repouso e em trânsito com **KMS** e **TLS**.

Esta arquitetura foi projetada para garantir a alta segurança e desempenho da aplicação de e-commerce da Nova Tech, utilizando os melhores serviços da AWS.