Cybersecurity

Project 1: Interview and Resume Guidance

Congratulations on completing your first project!

- This first project covered a wide range of topics including:
  - Systems administration
  - Networking
  - Network security
  - Cryptography
  - Virtualization
  - Cloud deployment
  - Web development

- When networking and talking to potential employers, you should be able to showcase your great work done on this project to answer specific interview questions or demonstrate your skills within these domains

- The following guide will assist you in adding this project to your resume and preparing you for interview questions about your project.

Resume Guidance

Many IT and cybersecurity professionals showcase their projects by adding them to their resume. Follow the instructions below to add your project to your resume:

- On the top of your resume, next to your contact information/LinkedIn profile. Add your domain to your cyber blog website.

- List your project under the “work experience” or “accomplishments” section of your resume, with the following information:

Project 1: Securing Cloud Applications

Technologies Used: Azure: {Keyvaults, App Services, Front Door, WAF}
PHP, HTML, Docker, OpenSSL

- Developed and designed a cyber-blog web application using Azure's Cloud services and Docker

- Created and stored SSL certificates in Azure's Key Vault, and bound them to secure the web application.

- Protected the web application by utilizing Azure's Security features, such as Azure's Front Door, WAF, and Security Center.

Interview Guidance

Good interview responses can adhere to the following structure:

1. Restate the problem.

2. Provide a concrete example scenario.

3. Explain the solution requirements.

4. Explain the solution details.

5. Identify advantages and disadvantages of the solution.Including each of these components will ensure you prove your competency of subject matter and critical thinking.

Sample Interview Questions and Answers

1. **Describe/summarize your Project.**

   I used Microsoft Azure to build and host my own "cyber-blog" web application. I secured it with a SSL certificate, and applied Azure's security features to protect it.

2. **What were your project Requirements?**

   Our requirements included:
   - Hosting the web application using Azure's Cloud Services

- Using Azure's App Service resource to create the web application
- Choosing a domain with "godaddy or Azure" (choose what you selected)
- Deploying a Docker container which had a framework for a blog webpage
- SSHing into the container to customize the webpage
- Creating a Self-signed certificate with OpenSSL
- Storing the certificate in Azure's Key vault
- Binding the certificate to the website (If you didn't select the free option)
- After determining the security issues with a self-signed certificate, creating and bound a managed CA approved certificate to the web application
- Deploying Azure's Front Door, and configuring a WAF rule to restrict traffic from certain countries
- Analyzing the Azure's Security Center recommendations and applying the recommend fix

3. **What were the advantages for choosing Azure's App Service resource, instead of creating a Virtual Machine From scratch?**

    - Using Azure App Service, we can pass responsibility of managing features outside of the web application to the cloud service provider
        - In other words, we are only responsible for deploying and managing their web application and it's associated data
    - Deploying web applications can be done much faster, as user's don't have to be concerned about configuring their OS
    - The user doesn't have to worry about the OS and middleware maintenance, such as installing software updates and patching
    - Azure App services are cheaper to run than Virtual machines
    - Azure App services have built in features for securing and hosting a web application, such as DNS, Web App Firewalls, Domain purchasing, SSL certification binding.

4. **What were the security issues you encountered when you used a self-signed certificate?**

    A Self-signed certificate is a certificate that has not been signed by a certificate authority. While these certificates are simple to create, and have no expense almost all browsers will alert the user visiting the webpage that is signed with a self-signed certificate that the webpage is not trusted by a Certificate Authority in the browser's root store and to proceed with caution.

5. **How did you address that security concern?**

I created and bound a managed certificate provided by Azure, as this is a trusted certificate which has been approved by a CA which most browsers trust.

6. **What is Azure's Front Door, and how did you use its WAF feature?**

   Azure's Front Door is a Cloud Resource, it resides in front of my web application to protect it. It works on the Application Layer of the OSI Model (Layer 7). Its primary solution is a load balancer. It can incorporate a Web Application Firewall (WAF) to protect against web vulnerabilities attacks. With the WAF, I applied a custom rule to protect against web requests from countries that I wasn't expecting any traffic from. This could help protect against potential attacks where I knew I wasn't expecting any visitors.

7. **What is Azure's Security Center, and how did you apply its features?**

   Azure Security Center is a management system that provides best practices and recommendations to enhance the security of your cloud resources. When I checked it, it had several security recommendations with various criticalities. One recommendation was to require FTP access into the web application with FTPS, which is the encrypted version of FTP.