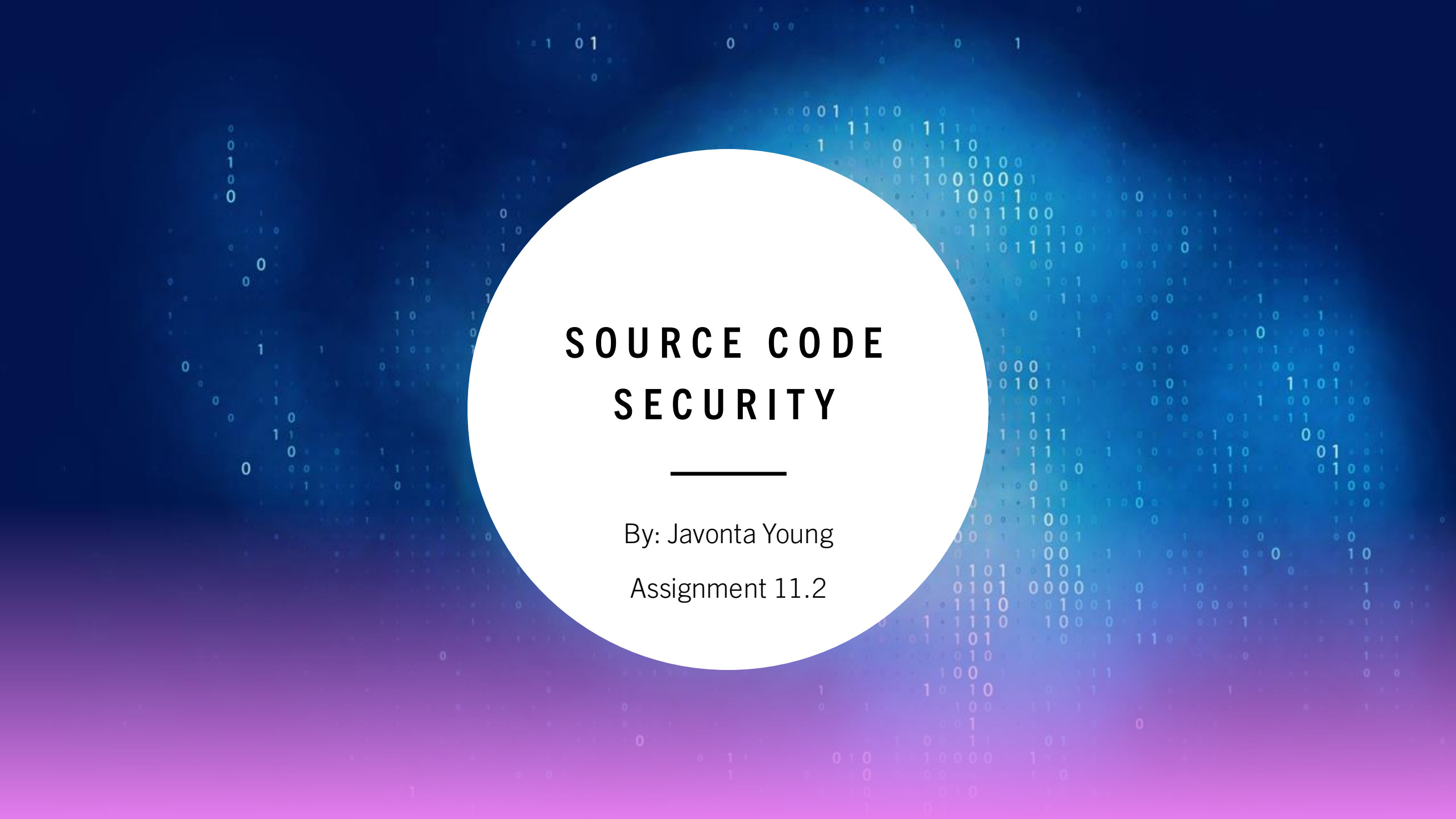


SOURCE CODE SECURITY

By: Javonta Young

Assignment 11.2

WHY DO YOU NEED SOURCE CODE PROTECTION

Your source code can contain secrets, such as API or encryption keys, OAuth tokens, passwords, and more. It is also common for PII to coexist with source code. Without protection, these are available to all repository contributors meaning that they can clone, copy, and distribute them.

Organizations must protect their valuable source code from various security risks, including outsider and insider threats. If it gets leaked or stolen, source code may not only give your competitors a leading edge in the development of new products, causing financial damage to your business, but hackers can also use it to exploit vulnerabilities. Besides competitive and financial damage, it can even ruin your company if it falls into the wrong hands.

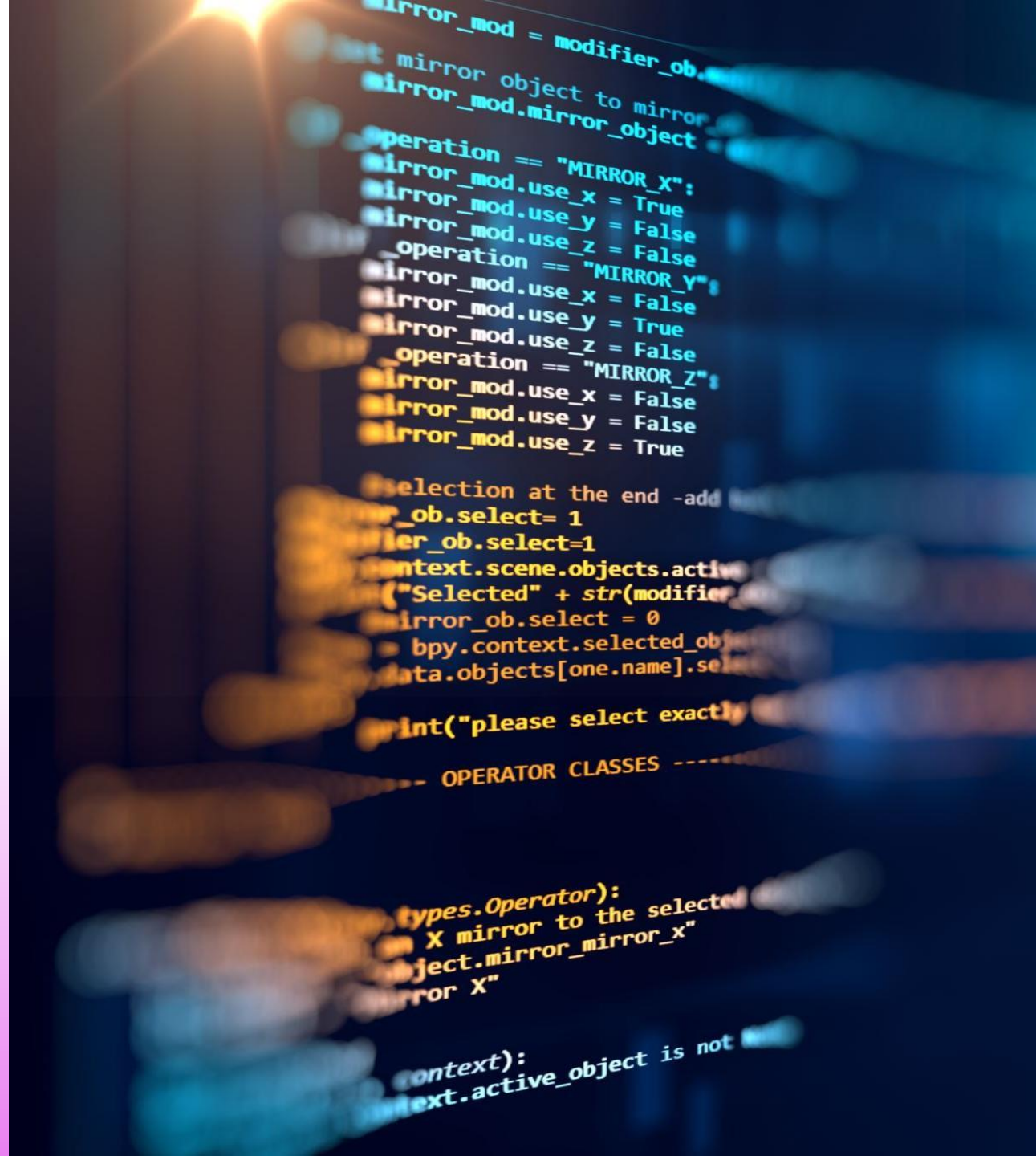
Source code security is vital to the health of your organization, especially if you balance the potential risks and business impacts that security vulnerabilities can have.

CREATE A SOURCE CODE PROTECTION POLICY

Set up a source code protection policy by defining a set of rules, requirements, and procedures for handling and protecting code. This policy will help safeguard software and devices from threats such as reverse engineering and code tampering. It should also cover source code development processes and personnel involved in code development.

Include secure access and use of source code repositories such as Git and Apache Subversion, encryption protocols, application hardening, shielding processes, and in-app protection methods.

Your source code protection policy should also involve documentation and training on secure coding practices and the incorporation of secure development methodologies into the software development lifecycle (SDLC).



PREVENT THE USE OF INSECURE SOURCE CODE

Use source code security analysis tools, such as Static Application Security Testing (SAST), to detect security flaws and other issues during development.

Static code analyzers scan source code and related dependencies (frameworks and libraries) for specific vulnerabilities as well as for compliance with coding standards. These tools reduce security risks in applications by finding vulnerabilities earlier in the SDLC and providing real-time feedback to development teams on issues.

SAST tools, however, cannot identify vulnerabilities outside the code, such as those defects that might be found in third-party interfaces. For this, you'll need Dynamic Application Security Testing (DAST) tools that can detect a wide range of vulnerabilities, including the ones from the OWASP Top Ten. Examples include cross-site scripting (XSS), injection errors like SQL injection, path traversal, and insecure server configuration.

ACCESS CONTROL

Define who's allowed to access source code, codebase and source code repositories. There's little to no reason that anyone other than hands-on employees work with your source code, but even for those that do, set up two-factor authentication. In this way, you can ensure that no suspicious characters find their way into your source code. Through authentication and authorization, access control policies ensure that users are who they say they are and that they have appropriate access to company data.

USE ENCRYPTION AND MONITORING

Make sure you have the ability to encrypt sensitive data both in transit and at rest. It's also important to monitor your data at all times and be alerted when any suspicious activity comes to light. In this way, you can be ready to act swiftly, whether it is about tracking, limiting, or reversing the damage. You can also prevent it before any actual harm happens.

DEPLOY NETWORK SECURITY TOOLS

Implementing network security solutions such as firewalls, Virtual Private Networks (VPN), anti-virus, and anti-malware software count as basic protection. These solutions safeguard your source code from external exploits of hackers and ensure secure data sharing between employees and data sources.

DON'T FORGET ABOUT ENDPOINT SECURITY

Secure your endpoints or entry points of end-user devices such as desktops and laptops from risky activities and malicious attacks with endpoint security software. Data Loss Prevention (DLP) solutions can efficiently prevent your source code from leaving the endpoint and stop source code exfiltration.

These tools can protect sensitive information both in physical and virtual environments, regardless of the endpoint's physical location and whether it's connected to the internet or not. Endpoint DLPs offer you the possibility to track the movement of sensitive data and take remediation actions.

RESOURCES

https://www.endpointprotector.com/blog/your-ultimate-guide-to-source-code-protection/