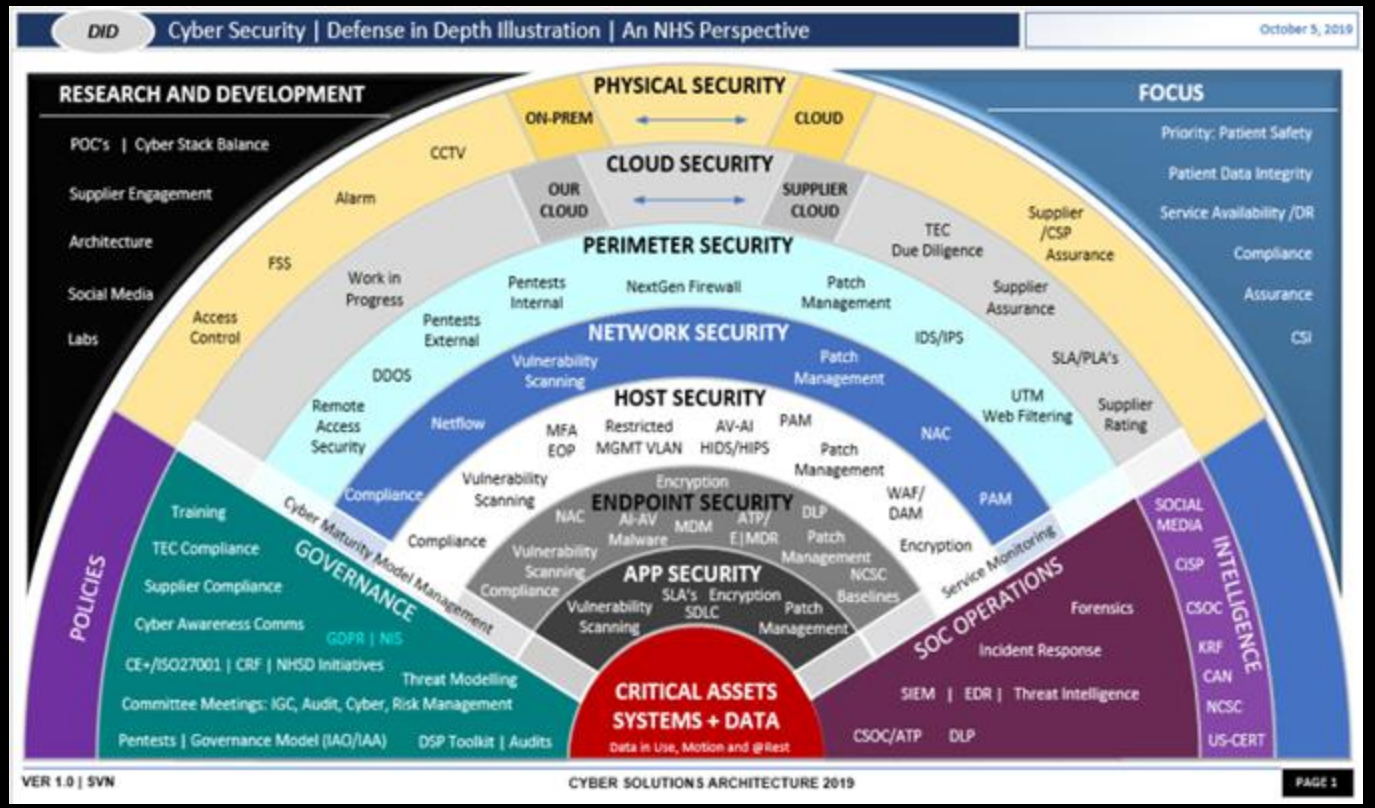
Green Pace

Security Policy Presentation

Developer: Bobby Rust

OVERVIEW: DEFENSE IN DEPTH



Green Pace

THREATS MATRIX

Likely STD-002-CPP Prevents out-of-bounds indexing of arrays which gives unwarranted access to memory.	**Priority** STD-004-CPP Prevents SQL injection by separating SQL commands from data.
Low priority STD-003-CPP Mitigate buffer overflow risk by validating the size of strings before writing to their memory.	**Unlikely** STD-006-CPP Assertions should be used only for debugging purposes. See https://cwe.mitre.org/data/definitions/617.html.

Green Pace

10 PRINCIPLES

1. Validate input
2. Heed compiler warnings
3. Architect and design for security policies
4. Keep it simple
5. Default deny
6. Adhere to the principle of least privilege
7. Sanitize data sent by other systems
8. Practice defense in depth
9. Use effective quality assurance techniques
10. Adopt a secure coding standard

CODING STANDARDS

- Data value
- String correctness
- SQL injection
- Memory protection
- Cryptographic algorithms
- Environment variables
- Update dependencies
- Exceptions
- Data types
- Assertions

Green Pace

ENCRYPTION POLICIES

- **Data at rest**
  - Use strong encryption algorithms.
  - Implement access control (role based)
- **Data in Transit**
  - Use HTTPS
  - VPNs, SSH

- **Data in use**
  - Secure coding practices

Green Pace

TRIPLE-A POLICIES

- Authentication Policies
  - Enforce strict password requirements, including length and complexity.
  - Mandate all organization accounts to use multi-factor authentication.
- Authorization Policies
  - Implement role-based access controls
  - Grant the least number of privileges necessary to each role
- Accounting Policies
  - Utilize logging software to maintain historical data of all network activity.
  - Realtime alerts for suspicious activity.

Green Pace

Unit Testing

The String Correctness standard is an extremely important standard. If strings are not handled properly, the code is vulnerable

Unsafely setting a string's value

Noncompliant Code

`strcpy` can write arbitrary values to memory if the source string is too long to fit into the destination string.

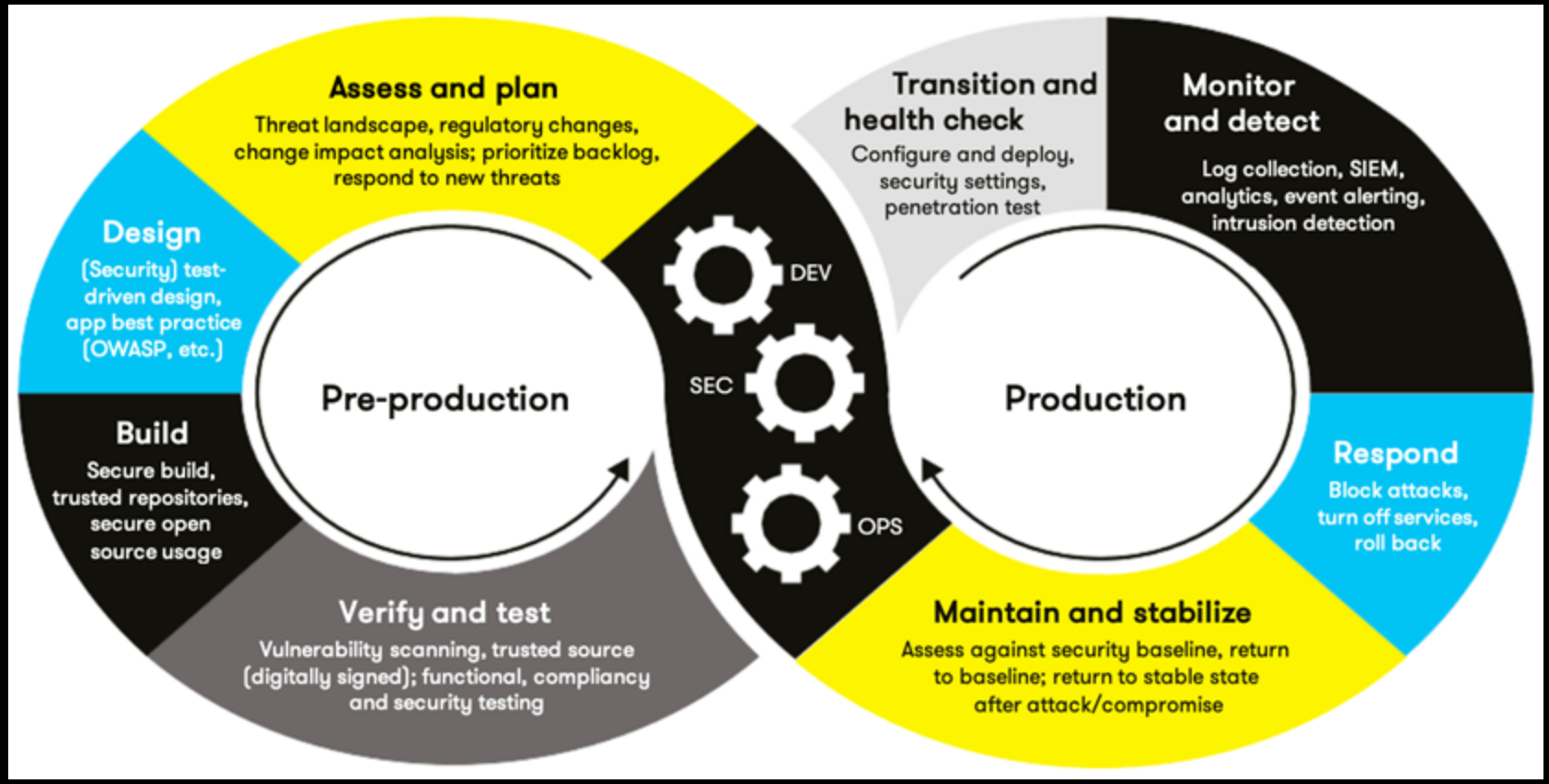
```
char dest[10];
strcpy(dest, src);
```

Safely setting a string's value

`strncpy` only writes the first `n` characters of the source string.

```
char dest[10];
strncpy(dest, source, sizeof(dest) - 1);
dest[sizeof(dest) - 1] = '\0';
```

AUTOMATION SUMMARY



Green Pace

TOOLS

- During the planning and design phases
  - Automation tools can be used to scan vulnerability databases.
- During the build phase
  - External tools such as CppCheck are used to check code for security vulnerabiltlies.
- Verification and Testing
  - Unit tests
- Production phase
  - Logging
  - Detection
  - Response

Green Pace

RISKS AND BENEFITS

- Problems
  - Phishing, ransomware, data breaches
  - Need to comply with regulations such as HIPAA, GDPR
  - Employee Awareness
  - Lack of incident response policies
- Solutions
  - Following a robust security policy
  - Provide regular employee training to increase security awareness
- Risks and Benefits
  - Acting now costs money, time, and resources in the short term with the benefit of reduced risk of security breaches, which can be even more costly
  - Waiting increases the risk of an attack in the short term, but gives time to think through a viable solution

Green Pace

RECOMMENDATIONS

- Security gaps and future improvements
  - Lack of standardization of employee training programs
  - No incident response plan
- In 2021, LinkedIn suffered a major data breach
  - 92% of user data became publicly available
  - Email addresses, phone numbers, job titles, geolocation data
  - Could have been prevented with a more secure API

Green Pace

CONCLUSIONS

- Employee training programs
- Incident response plans
- API security standardization, including robust authentication and authorization
- Cloud security standards
- Increase awareness of AI assisted attack vectors, which improve the quality of phishing attacks

Green Pace

REFERENCES

- B. Gibson, S. Townes, D. Lewis and S. Bhunia, "Vulnerability in Massive API Scraping: 2021 LinkedIn Data Breach," *2021 International Conference on Computational Science and Computational Intelligence (CSCI)*, Las Vegas, NV, USA, 2021, pp. 777-782, doi: 10.1109/CSCI54926.2021.00191.

- One, A. (1996). *Smashing The Stack For Fun And Profit*. https://inst.eecs.berkeley.edu/~cs161/fa08/papers/stack_smashing.pdf

Green Pace