IoT Security Guidance

From OWASP

Back To The Internet of Things Project (https://www.owasp.org/index.php/OWASP_Internet_of_Things_Project)

Manufacturer IoT Security Guidance

(DRAFT)

The goal of this section is help manufacturers build more secure products in the Internet of Things space. The guidance below is at a basic level, giving builders of products a basic set of guidelines to consider from their perspective. This is not a comprehensive list of considerations, and should not be treated as such, but ensuring that these fundamentals are covered will greatly improve the security of any IoT product.

Category	IoT Security Consideration
I1: Insecure Web Interface	▪ Ensure that any web interface in the product disallows weak passwords ▪ Ensure that any web interface in the product has an account lockout mechanism ▪ Ensure that any web interface in the product has been tested for XSS, SQLi and CSRF vulnerabilities ▪ Ensure that any web interface has the ability to use HTTPS to protect transmitted information ▪ Include web application firewalls to protect any web interfaces ▪ Ensure that any web interface allows the owner to change the default username and password
I2: Insufficient Authentication/Authorization	▪ Ensure that any access requiring authentication requires strong passwords ▪ Ensure that user roles can be properly segregated in multi-user environments ▪ Implement two-factor authentication where possible ▪ Ensure password recovery mechanisms are secure ▪ Ensure that users have the option to require strong passwords ▪ Ensure that users have the option to force password expiration after a specific period ▪ Ensure that users have the option to change the default username and password
I3: Insecure Network Services	▪ Ensure all devices operate with a minimal number of network ports active ▪ Ensure all devices do not make network ports and/or services available to the internet via UPnP for example ▪ Review all required network services for vulnerabilities such as buffer overflows or denial of service
I4: Lack of Transport Encryption	▪ Ensure all communication between system components is encrypted as well as encrypting traffic between the system or device and the internet ▪ Use recommended and accepted encryption practices and avoid proprietary protocols ▪ Ensure SSL/TLS implementations are up to date and properly configured ▪ Consider making a firewall option available for the product
I5: Privacy Concerns	▪ Ensure only the minimal amount of personal information is collected from consumers ▪ Ensure all collected personal data is properly protected using encryption at rest and in transit ▪ Ensure only authorized individuals have access to collected personal information ▪ Ensure only less sensitive data is collected ▪ Ensuring data is de-identified or anonymized ▪ Ensuring a data retention policy is in place ▪ Ensuring end-users are given a choice for data collected beyond what is needed for proper operation of the device
I6: Insecure Cloud Interface	▪ Ensure all cloud interfaces are reviewed for security vulnerabilities (e.g. API interfaces and cloud-based web interfaces) ▪ Ensure that any cloud-based web interface disallows weak passwords ▪ Ensure that any cloud-based web interface has an account lockout mechanism ▪ Implement two-factor authentication for cloud-based web interfaces ▪ Ensure that all cloud interfaces use transport encryption ▪ Ensure that any cloud-based web interface has been tested for XSS, SQLi and CSRF vulnerabilities ▪ Ensure that users have the option to require strong passwords ▪ Ensure that users have the option to force password expiration after a specific period ▪ Ensure that users have the option to change the default username and password
I7: Insecure Mobile Interface	▪ Ensure that any mobile application disallows weak passwords ▪ Ensure that any mobile application has an account lockout mechanism ▪ Implement two-factor authentication for mobile applications (e.g Apple's Touch ID) ▪ Ensure that any mobile application uses transport encryption ▪ Ensure that users have the option to require strong passwords ▪ Ensure that users have the option to force password expiration after a specific period

	▪ Ensure that users have the option to change the default username and password
I8: Insufficient Security Configurability	▪ Ensure password security options are made available (e.g. Enabling 20 character passwords or enabling two-factor authentication) ▪ Ensure encryption options are made available (e.g. Enabling AES-256 where AES-128 is the default setting) ▪ Ensure secure logging is available for security events ▪ Ensure alerts and notifications are available to the user for security events
I9: Insecure Software/Firmware	▪ Ensure all system devices have update capability and can be updated quickly when vulnerabilities are discovered ▪ Ensure update files are encrypted and that the files are also transmitted using encryption ▪ Ensure that update files are signed and then validated by the device before installing ▪ Ensure update servers are secure ▪ Ensure the product has the ability to implement scheduled updates
I10: Poor Physical Security	▪ Ensure the device is produced with a minimal number of physical external ports (e.g. USB ports) ▪ Ensure the firmware of Operating System can not be accessed via unintended methods such as through an unnecessary USB port ▪ Ensure the product is tamper resistant ▪ Ensure the product has the ability to limit administrative capabilities in some fashion, possibly by only connecting locally for admin functions ▪ Ensure the product has the ability to disable external ports such as USB

General Recommendations

Consider the following recommendation for all Internet of Things products:

- Avoid the potential for persistent vulnerabilities in devices that have no update capability by ensuring that all devices and systems are built with the ability to be updated when vulnerabilities are discovered
- Rebranded devices used as part of a system should be properly configured so that unnecessary or unintended services do not remain active after the rebranding

[NOTE: Given the fact that each deployment and every environment is different, it is important to weigh the pros and cons of implementing the advice above before taking each step.]

Developer IoT Security Guidance

(DRAFT)

The goal of this section is help developers build more secure applications in the Internet of Things space. The guidance below is at a basic level, giving developers of applications a basic set of guidelines to consider from their perspective. This is not a comprehensive list of considerations, and should not be treated as such, but ensuring that these fundamentals are covered will greatly improve the security of any IoT product. Strongly consider using a Secure IoT Framework (https://www.owasp.org/index.php /OWASP_Internet_of_Things_Project#tab=IoT_Framework_Assessment) in order to proactively address many of the concerns listed below.

Category	IoT Security Consideration	Recommendations
I1: Insecure Web Interface	▪ Ensure that any web interface coding is written to prevent the use of weak passwords ▪ Ensure that any web interface coding is written to include an account lockout mechanism ▪ Ensure that any web interface coding has been tested for XSS, SQLi and CSRF vulnerabilities ▪ Ensure that any web interface has the ability to use HTTPS to protect transmitted information ▪ Ensure that any web interface coding is written to allow the owner to change the username and password ▪ Consider the use of web application firewalls to protect any web interfaces	When building a web interface consider implementing lessons learned from web application security. Employ a framework (https://www.owasp.org/index.php/OWASP_Periodic_Table_of_Vulnerabilities#Generic_Application_Frameworks) that utilizes security controls to ensure that vulnerabilities are mitigated in code. Be sure to plan for eventual upgrades or security fixes to the framework as well. If you use optional plugins to the framework be sure to review them for security. Deploy and protect the web interface in the same way you would any web application. Utilize encrypted transport protocols if possible, being sure to validate certificates. Limit access in whatever ways possible. Assume users will not change configuration so deploy in a secure manner with strong credentials already in place.
I2: Insufficient Authentication/Authorization	▪ Ensure that applications are written to require strong passwords where authentication is needed ▪ Ensure the application takes into account multi-user environments and includes functionality for role separation ▪ Implement two-factor	Refer to the OWASP Authentication Cheat Sheet (https://www.owasp.org/index.php/Authentication_Cheat_Sheet)

	authentication where possible ▪ Ensure password recovery mechanisms are written to function in a secure manner ▪ Ensure that applications are written to include the option to require strong passwords ▪ Ensure that applications are written to include the option to force password expiration after a specific period ▪ Ensure that applications are written to include the option to change the default username and password	
I3: Insecure Network Services	▪ Ensure applications that use network services don't respond poorly to buffer overflow, fuzzing or denial of service attacks ▪ Ensure applications test ports are taken out of service before going to production	Try to utilize tested, proven, networking stacks and interfaces that handle exceptions gracefully. Be sure that any test or maintenance interfaces are disabled or properly protected. Avoid exposing unauthenticated protocols (such as TFTP) or unencrypted channels (such as telnet) if possible. Consider the attack surface that device network services present. Turn off unnecessary services and deploy measures to protect required services, detect malicious activity, and react to an attack with measures such as lock-outs or temporary firewall rules.
I4: Lack of Transport Encryption	▪ Ensure all applications are written to make use of encrypted communication between devices and between devices and the internet ▪ Use recommended	Utilize encrypted protocols wherever possible to protect all data in transit. Where protocol encryption is not possible consider encrypting data before transfer.

	and accepted encryption practices and avoid proprietary protocols ▪ Consider making a firewall option available for the application	
I5: Privacy Concerns	▪ Ensure only the minimal amount of personal information is collected from consumers ▪ Ensure all collected personal data is properly protected using encryption at rest and in transit ▪ Ensuring data is de-identified or anonymized ▪ Ensuring end-users are given a choice for data collected beyond what is needed for proper operation of the device	Data can present unintended privacy concerns when aggregated. As a rule collect the minimal amount of data possible. Consult with data scientists, legal and compliance teams to determine risk of data collection and storage. Consider implications of consent and the fact that IoT devices may not present an interface for collecting consent and may passively collect data about people other than owners and operators. IoT may collect information about individuals who cannot provide consent (such as minors) and data collection should be modified accordingly. Also refer to the OWASP Top 10 Privacy Risks (https://www.owasp.org/index.php/OWASP_Top_10_Privacy_Risks_Project).
I6: Insecure Cloud Interface	▪ Ensure all cloud interfaces are reviewed for security vulnerabilities (e.g. API interfaces and cloud-based web interfaces) ▪ Ensure that any cloud-based web interface coding is written to disallows weak passwords ▪ Ensure that any cloud-based web interface coding is written to include an account lockout mechanism ▪ Implement	Cloud security presents unique security considerations, as well as countermeasures. Be sure to consult your cloud provider about options for security mechanisms. Consult the OWASP Cloud Top 10 Security Risks (https://www.owasp.org/index.php/Category:OWASP_Cloud_%E2%80%90_10_Project) documents.

	two-factor authentication for cloud-based web interfaces ▪ Ensure that any cloud interface coding has been tested for XSS, SQLi and CSRF vulnerabilities ▪ Ensure that all cloud interfaces use transport encryption ▪ Ensure that cloud interfaces are written to include the option to require strong passwords ▪ Ensure that cloud interfaces are written to include the option to force password expiration after a specific period ▪ Ensure that cloud interfaces are written to include the option to change the default username and password	
I7: Insecure Mobile Interface	▪ Ensure that any mobile application coding is written to disallows weak passwords ▪ Ensure that any mobile application coding is written to include an account lockout mechanism ▪ Implement two-factor authentication for mobile applications (e.g Apple's Touch ID)	Mobile interfaces to IoT ecosystems require targeted security. Consult the OWASP Mobile Project (https://www.owasp.org/index.php /OWASP_Mobile_Security_Project) for further guidance.

	▪ Ensure that any mobile application uses transport encryption ▪ Ensure that mobile interfaces are written to include the option to require strong passwords ▪ Ensure that mobile interfaces are written to include the option to force password expiration after a specific period ▪ Ensure that mobile interfaces are written to include the option to change the default username and password ▪ Ensure that mobile interfaces only collect the minimum amount of personal information needed	
I8: Insufficient Security Configurability	▪ Ensure applications are written to include password security options (e.g. Enabling 20 character passwords or enabling two-factor authentication) ▪ Ensure applications are written to include encryption options (e.g. Enabling AES-256 where AES-128 is the default setting) ▪ Ensure all	Security can be a value proposition. Design should take into consideration a sliding scale of security requirements. Architect projects with secure defaults and allow consumers to select options to be enabled or disabled. IoT design should be forward compatible with respect to security - as cipher suites increase and new security technologies become widely available IoT design should be able to adopt these new technologies. Remember the security lifecycle of protect, detect, and react. Design systems to allow for the detection of malicious activity as well as self defending capabilities and a reaction plan should a compromise be detected. Design all stages of the lifecycle to be evolutionary so improvements can be added to a system or device future releases, updates, or patches.

	applications are written to produce logs for security events ▪ Ensure all applications are written to produce alerts and notifications to the user for security events	
I9: Insecure Software/Firmware	▪ Ensure all applications are written to include update capability and can be updated quickly when vulnerabilities are discovered ▪ Ensure all applications are written to process encrypted update files and that the files are transmitted using encryption ▪ Ensure all applications are written to process signed files and then validate that file before installation	Many IoT deployments are either brownfield (i.e. applied over existing infrastructure) and/or have an extremely long deployment cycle. To maintain the security of devices over time it is critical to plan for patches and updates. Confidentiality, Integrity, and Availability (CIA) are primary concerns when providing binaries and updates to edge devices. Encrypt updates before distribution, providing decryption keys along with download instructions to authorized devices. Updates should have cryptographic signatures using public key cryptography that can be verified by devices. A cryptographic signature allows for distribution of updates over untrusted channels, such as Content Delivery Network (CDN), peer-to-peer, or machine to machine (M2M). Devices should always validate cryptographic certificates and discard updates that are not properly delivered or signed. If unencrypted updates are utilized be sure that a cryptographic hash of the update is provided over an encrypted channel so the device can detect tampering. Provide a mechanism for issuing, updating and revoking cryptographic keys as well. Key management and lifecycle should be taken into consideration prior to deployment. This includes the SSL trust store, or root trust, on a device, which may have to be modified over the lifespan of the device.
I10: Poor Physical Security	▪ Ensure applications are written to utilize a minimal number of physical external ports (e.g. USB ports) on the device ▪ Ensure all applications can not be accessed via unintended methods such as through an unnecessary USB port ▪ Ensure all applications are written to allow for	Plan on having IoT edge devices fall into malicious hands. Utilize whatever physical security protections are available. Disable any testing or debugging interfaces, utilize Hardware Security Modules (HSM's), cryptographic co-processors, and Trusted Platform Modules (TPM's) wherever possible. Consider the implications of a compromised device. Do not share credentials, application or cryptographic keys across multiple devices to limit the scope of damage due to a physical compromise. Plan for the transfer of ownership of devices and ensure that data is not transferable along with the ownership.

	disabling of unused physical ports such as USB ▪ Consider writing applications to limit administrative capabilities to a local interface only	

General Recommendations

Consider the following recommendations for all user interfaces (local device, cloud-based and mobile):

- Avoid potential Account Harvesting issues by:
  - Ensuring valid user accounts can't be identified by interface error messages
  - Ensuring strong passwords are required by users
  - Implementing account lockout after 3 - 5 failed login attempts

[NOTE: Given the fact that each deployment and every environment is different, it is important to weigh the pros and cons of implementing the advice above before taking each step.]

Consumer IoT Security Guidance

(DRAFT)

The goal of this section is help consumers purchase secure products in the Internet of Things space. The guidance below is at a basic level, giving consumers a basic set of guidelines to consider from their perspective. This is not a comprehensive list of considerations, and should not be treated as such, but ensuring that these fundamentals are covered will greatly aid the consumer in purchasing a secure IoT product.

Category	IoT Security Consideration
I1: Insecure Web Interface	▪ If your system has the option to use HTTPS, ensure it is enabled ▪ If your system has a two factor authentication option, ensure that it is enabled ▪ If your system has web application firewall option, ensure that it is enabled ▪ If your system has a local or cloud-based web application, ensure that you change the default password to a strong one and if possible change the default username as well ▪ If the system has account lockout functionality, ensure that it is enabled ▪ Consider employing network segmentation technologies such as firewalls to isolate IoT systems from critical IT systems
I2: Insufficient Authentication/Authorization	▪ If your system has a local or cloud-based web application, ensure that you change the default password to a strong one and if possible change the default username as well ▪ If the system has account lockout functionality, ensure that it is enabled ▪ If the system has the option to require strong passwords, ensure that is enabled ▪ If the system has the option to require new passwords after 90 days for example, ensure that is enabled ▪ If your system has a two factor authentication option, ensure that it is enabled ▪ If your system has the option to set user privileges, consider setting user privileges to the minimal needed for operation ▪ Consider employing network segmentation technologies such as firewalls to isolate IoT systems from critical IT systems
I3: Insecure Network Services	▪ If your system has a firewall option available, enable it and ensure that it can only be accessed from your client systems ▪ Consider employing network segmentation technologies such as firewalls to isolate IoT systems from critical IT systems
I4: Lack of Transport Encryption	▪ If your system has the option to use HTTPS, ensure it is enabled
I5: Privacy Concerns	▪ Do not enter sensitive information into the system that is not absolutely required, e.g. address, DOB, CC, etc. ▪ Deny data collection if it appears to be beyond what is needed for proper operation of the device (If provided the choice)
I6: Insecure Cloud Interface	▪ If your system has the option to use HTTPS, ensure it is enabled ▪ If your system has a two factor authentication option, ensure that it is enabled ▪ If your system has web application firewall option, ensure that it is enabled ▪ If your system has a local or cloud-based web application, ensure that you change the default password to a strong one and if possible change the default username as well ▪ If the system has account lockout functionality, ensure that it is enabled ▪ If the system has the option to require strong passwords, ensure that is enabled ▪ If the system has the option to require new passwords after 90 days for example, ensure that is enabled
I7: Insecure Mobile Interface	▪ If the mobile application has the option to require a PIN or password, consider using it for extra security (on client and server) ▪ If the mobile application has the option to use two factory authentication such as Apple's Touch ID, ensure it is enabled ▪ If the system has account lockout functionality, ensure that it is enabled ▪ If the system has the option to require strong passwords, ensure that is enabled ▪ If the system has the option to require new passwords after 90 days for example, ensure that is enabled ▪ Do not enter sensitive information into the mobile application that is not absolutely required, e.g. address, DOB, CC, etc.
I8: Insufficient Security Configurability	▪ If your system has the option, enable any logging functionality for security-related events ▪ If your system has the option, enable any alert and notification functionality for security-related events

	▪ If your system has security options for passwords, ensure they are enabled for strong passwords ▪ If your system has security options for encryption, ensure they are set for an accepted standard such as AES-256
I9: Insecure Software/Firmware	▪ If your system has the option to verify updates, ensure it is enabled ▪ If your system has the option to download updates securely, ensure it is enabled ▪ If your system has the ability to schedule updates on a regular cadence, consider enabling it
I10: Poor Physical Security	▪ If your system has the ability to limit administrative capabilities possible by connecting locally, consider enabling that feature ▪ Disable any unused physical ports through the administrative interface

General Recommendations

If you are looking to purchase a device or system, consider the following recommendations:

- Include security in feature considerations when evaluating a product
- Place Internet of Things devices on a separate network if possible using a firewall

[NOTE: Given the fact that each deployment and every environment is different, it is important to weigh the pros and cons of implementing the advice above before taking each step.]

Retrieved from "https://wiki.owasp.org/index.php?title=IoT_Security_Guidance&oldid=226350"
