P1 Web Application Vulnerabilities	Vulnerability is a key problem in any system that guards or operates on sensitive user data. Failure to suitably design and implement an application, detect a problem or promptly apply a fix (patch) is likely to result in a privacy breach. This risk also encompasses the OWASP Top 10 List of web application vulnerabilities and the risks resulting from them.
How to check? ● Are regular penetration tests performed with a focus on privacy? ● Are developers trained regarding web application security? ● Are secure coding guidelines applied? ● Is any of the used software out of date (server, database, frameworks, other infrastructure components)?	Countermeasures ● Perform regular penetration tests by independent security experts. ● Track remediation of findings. ● Train application developers and architects in secure development. ● Apply procedures for secure development (e.g. Security Development Lifecycle - SDL). ● Install updates, patches and hotfixes on a regular basis.
Example ● Injection Flaws allow attackers among others to copy or manipulate data by attacks like SQL injection. ● Sensitive Data Exposure allows attackers gather sensitive information e.g. due to missing encryption with a man-in-the-middle attack. ● Use of Insecure Direct Object References allows attackers to guess and access sensitive information, especially if access control is missing. ● Usage of Components with Known Vulnerabilities, e.g. unpatched software flaws, and Security Misconfigurations, e.g. unhardened application platform. ● In general it is possible for attackers to gain access to, manipulate or delete personal data that the application is processing by abusing rights, entering malicious code or eavesdropping on communications.	References ● OWASP Top 10 Project ● OWASP ASVS ● Open SAMM ● OWASP Proactive Controls ● Security Development Lifecycle (SDL) ● OWASP Secure Application Design Project ● Lists of known vulnerabilities can be found at CVE and NVD ● ISMS of the German Federal Office for Information Security (BSI)

P2 Operator-sided Data Leakage	Failure to prevent the leakage of any information containing or related to user data, or the data itself, to any unauthorized party resulting in loss of data confidentiality. Introduced either due to intentional malicious breach or unintentional mistake e.g. caused by insufficient access management controls, insecure storage, duplication of data or a lack of awareness.
How to check?	Countermeasures
● Research the reputation and reliability of the operator: ○ Have there been former breaches related to the operator? ○ Does the provider proactively prove privacy and security and if yes, how? ○ Is there a bug bounty program to report vulnerabilities? ○ Is the provider certified according to ISO 27001 or ISO 27018 (cloud providers)? ○ Is the operator located in a country with high privacy standards? ● Audit the operator: ○ Are privacy best practices in place? ○ Is awareness training mandatory for all employees? ○ Is there a privacy engineering team? ○ How is personal data anonymized? ○ Is personal data encrypted? ○ Who has access to the data (need-to-know-principle)? ● Audit methods: ○ Paper-based audit (fair) ○ Interview-based audit (good) ○ On-site audit and system-checks (best)	● Appropriate Identity and Access Management (physical as well as logical): ○ Principle of least privilege. ● Use strong encryption for all personal data stored (data at rest) especially on mobile media (e.g. USB memory sticks, laptop hard disks, tablet and phone local storage, backup tapes, portable hard disk drives). ● Awareness training for all employees regarding handling of personal data. ● Implementation of a data classification and information handling policy. ● Monitor and detect classified data when it leaks from endpoints, web portals and cloud services (e.g. by Data Leakage Prevention, SIEM). ● Implement Privacy by Design ● Anonymisation of personal data: It is common practice to anonymise personal data and use it for other purposes e.g. testing or marketing. Anonymisation is not easy (e.g. aol search data leak) and there are many anonymisation theories which can be very complex. ● Pseudonymisation which means that data can only be connected to a person with help of a third party that knows the person and corresponding pseudonym.
Example	References
● Handbook for Safeguarding Sensitive PII	● Article 29 Working Party on Anonymization ● IT-Grundschutz-Catalogues

P3 Insufficient Data Breach Response	Not informing the affected persons (data subjects) about a possible breach or data leak, resulting either from intentional or unintentional events; failure to remedy the situation by fixing the cause; not attempting to limit the leaks.
How to check?	Countermeasures
General questions: ● Is an incident response plan for privacy incidents in place? ● Is this plan tested regularly (provide evidence e.g. a test protocol)? ● Do you have a Computer Emergency Response Team (CERT) and / or a Privacy Team? ● Do you have monitoring for incidents (e.g. SIEM) in place? If there was a privacy incident, did you: ● detect it (timeously)? ● notify relevant parties, including the individuals themselves, in a timely manner? ● protect evidence, remaining data during response / investigation? Is your incident response: ● Timely - information is disclosed to affected parties soon enough for them to avoid additional harm? ● Honest, accurate and understandable? Organizations that experience a privacy breach have a responsibility to clearly communicate the nature and scope of the breach to those affected. ● Established company wide for security breach notifications (policy)?	Countermeasures (in advance): ● Create and maintain incident response plan. ● Test incident response plan regularly. ● Include privacy-related incidents in test. ● Establish a Computer Emergency Response Team (CERT). ● Establish a Privacy Team. ● Continuously monitor for personal data leakage and loss. Responding to the breach: ● Validate the breach. ● Once a breach has been validated, immediately assign an incident manager to be responsible for the investigation. ● Assemble incident response team. ● Determine the scope and composition of the breach (e.g. legislation, confidentiality). ● Notify the data owners. ● Determine whether to notify the authorities (situation dependent). ● Decide how to investigate the data breach to ensure that the evidence is appropriately handled. ● Determine whether notification of affected individuals is appropriate and if so, when and how. ● Collect and review any breach response documentation and analyse reports.
Example	References
AICPA Privacy Incident Response Plan Template ENISA recommendations for severity assessment	Key Steps for Organizations in Responding to Privacy Breaches (Privacy Commissioner of Canada) Data Breach Response Checklist (PTAC)

P4 Insufficient Deletion of Personal Data	Failure to effectively and / or timeously delete personal data after termination of the specified purpose or upon request.
How to check? ● Inspect the data retention / deletion policies and/or agreements. ● Evaluate their appropriateness. ● Request deletion protocols. ● Test processes for deletion requests. ● Check if transparency is provided (which data is deleted when and which data is not deleted and why).	Countermeasures ● Deploy systems with good privacy practices, in this case minimization. ● Personal data has to be deleted after termination of the specified purpose and after an appropriate timeframe (e.g. one month). ● Personal data has to be deleted on rightful user request. ● Secure locking (with very limited access to the data) might be an option if deletion is not possible due to technical restrictions. ● Real deletion is preferable though and minimizes the risk. ● Data retention, archival and deletion policies and processes have to be documented and followed. ● Evidence should be collected to verify the deletion as per policy. ● Any data in backups, other copies or shared with third parties has to be considered. ● Exceptions are possible in case of retention required by law. Access should be very limited and protocolled for this case. ● When deleting data in cloud, take note of historical data stored in older snapshots. ● Deletion of user profiles after longer periods of inactivity.
Example Customer data is deleted automatically after a certain period of inactivity (Hotmail removes user profiles in case they are not used for one year) or after termination of contract (it is not required by law to keep all customer information for accounting or other purposes).	References ● https://ico.org.uk/for-organisations/guide-to-data-protection/principle-5-retention/ ● German DIN standard 66398

P5 Non-transparent Policies, Terms and Conditions	Not providing sufficient information to describe how data is processed, such as its collection, storage, processing and deletion. Failure to make this information easily-accessible and understandable for non-lawyers.
How to check? Check if policies, terms and conditions: ● Are easy to find ● Fully describe data processing: ○ Who are you / who is processing the data ○ Including data transfers ○ Analysis performed ○ Retention time ○ Meta data used ○ What are the rights ○ … ● Understandable for non-lawyers ● Complete, but KISS (Keep it short and simple) ● Include a process for obtaining user consent if the terms, policies or conditions change. ● Are available in the user's language ● Explain which data are collected ● Explain the purposes for which personal data is collected ● Use a readability tester like https://readability-score.com/ to check whether a text is hard to read or not. ● Are privacy rules actively communicated or does the user have to take action	Countermeasures ● Terms & Conditions (T&Cs) should be specifically for the use and data processing of the website. ● They should be easy to understand for non-lawyers and not too long. ● Provide an easily readable summary of the terms and conditions as well as a long version. ● Pictograms can be used for visual aid. ● Use separate T&Cs for use and data processing. ● Use release notes to identify change history of T&Cs and policies/notices over time. ● Keep track of which users consented to which version and any other time at which they may opt in to newer versions. ● Deploy Do Not Track on the server side. ● When collecting information it should be clear why it is needed. You should also try to predict whether you will be likely to do other things with it in the future and tell the users if you have such plans. ● Provide a list of cookies, widgets etc. used with an explanation of the use e.g. sharing data or advertising. ● Provide an opt-out-button for the users.
Example ● Easily readable summaries: ○ http://www.avg.com/privacy ○ 500px.com ● Explanation of cookies, widgets etc. including an opt-out-button if existing: ○ http://www.kaspersky.com/third-party-tracking ● Examples for Pictograms: ○ http://netdna.webdesignerdepot.com/uploads/2014/03/iubenda.jpg	References ● Privacy notices code of practice from ICO, also contains a list of examples: https://ico.org.uk/media/for-organisations/documents/1610/privacy_notices_cop.pdf ● HTTPA (HTTP with Accountability) ● Biggest lie is a project that protests against overly complicated T&Cs and shows other projects that try to change that.

P6 Collection of data not required for the primary purpose	Collecting descriptive, demographic or any other user-related data that are not needed for the purposes of the system. Applies also to data for which the user did not provide consent.
How to check? ● List personal data collected by the application. ● Request description of purpose. ● Check if collected data is required to fulfill the purpose. ● If data is collected that is not required for the primary purpose(s), check if consent to collect and process this data was given and is documented. ● Are individuals notified and asked if purpose or processing is changed? ● Are regular compliance checks regarding the collection of personal data and user consent in place?	Countermeasures ● Define the purpose of the collection of personal data. ● Only collect personal data required to fulfill the purpose. ● Default is to collect as little data as possible unless the user chooses otherwise (data reduction / minimization). ● Provide the data subject the option to provide additional data voluntarily to improve the service (e.g. product recommendation, personalized advertisement) with possibility to opt-out. ● The purpose for collection of personal data collected is specified no later than at the time of data collection. ● Conditioned collection: Collect personal data only if they are really required for an used feature.
Example Positive: ● A webshop collects Email addresses to send an order confirmation to the buyer. This email address is not used to send news about products (another purpose) unless the user actively chooses this option (opt-in). Negative: ● Amazon provides personalized advertisement to its users. This can be disabled, but the default setting is on. From a privacy point of view it should be disabled by default and the user should opt-in to receive personalized product recommendations.	References Article 29 Working Party Opinion on Purpose Limitation Privacy Design Strategies: ● M. Colesky, J.-H. Hoepman, and C. Hillen. **A Critical Analysis of Privacy Design Strategies**. In *2016 International Workshop on Privacy Engineering – IWPE'16*, San Jose, CA, USA, May 26 2016. (to appear). ● J.-H. Hoepman. **Privacy Design Strategies**. In *IFIP TC11 29th Int. Conf. on Information Security (IFIP SEC 2014)*, pages 446-459, June 2-4 2014.

P7 Sharing of Data with Third Party	Providing user data to any third-party, without obtaining the user’s consent. Sharing results either due to transfer or exchanging for a monetary compensation or otherwise due to inappropriate use of third-party resources included in the web site like widgets (e.g. maps, social networks buttons), analytics or web bugs (e.g. beacons).
How to check? ● Is personal data transferred to third parties? ● Are third party solutions in use (plugins, buttons, maps, videos, advertising, etc.) and which ones? ● Is third party tracking disclosed (which third parties and what data). ● Can you provide a list of all third parties? ● Check each third party against each of the criteria in this document. ● Did you rate them regarding privacy? ● Is privacy and handling of personal data part of the contract and if yes, what restrictions are in place? ● Do you use privacy-friendly implementations of third party content (if available)? ● Do you use blacklists of third parties that are forbidden due to privacy concerns? ● Do you audit your third parties? ● If you transfer data to third-parties, or use third-party processing, is there a user consent for sharing data?	Countermeasures Personal data is often shared with third parties through the integration of third party content like user tracking code, advertising banners, social network buttons or videos, and third-party hosted JavaScript and style sheet libraries. The following measures should be considered for a privacy-friendly use of third party content: ● Use third party content only where it is required, not by default. ● Use your own server as a “proxy” for content. ● Deploy full Do Not Track, to the latest W3C standard. Prefer W3C standard over unofficial EFF one. ● Tokenisation or anonymisation (data masking) should be considered for use before sharing of data with a third party. ● Develop a Third Party Monitoring Strategy: ○ Gateway release for third party content (whitelist or blacklist). ○ Contractual arrangements regarding Policies, Data usage, etc. ○ Monitoring of user complaints.
Example Social network buttons do not transfer data unless they are clicked on: https://github.com/heiseonline/shariff Youtube provides the opportunity to enable a privacy-enhanced mode and only transfers personal data in case of a click.	References W3C Working Draft Tracking Compliance and Scope Attribute-based Credentials for Trust: https://abc4trust.eu/ https://en.wikipedia.org/wiki/Do_Not_Track

P8 Outdated personal data	The use of outdated, incorrect or bogus user data. Failure to update or correct the data.
How to check? ● Ask the operator how it is ensured that personal data is up-to-date. ● Check for possibilities to update personal data in the application. ● Are there regular checks to validate that data is up-to-date (e.g. “please verify your shipping address”)? ● Question how long it is likely that data is up to date and how often it usually changes.	Countermeasures ● Implement a procedure to update the user’s personal data by obtaining inputs from them after a certain time period. ● The user should approve data if he or she is triggering a “critical” action. ● Provide a form to enable users to update their data. ● In case of an update make sure to forward the information to any third parties / subsystems that received the user’s data before (if there are any).
Example An update form is provided on the website so that the user can update his or her data when needed. Amazon is asking whether your address and account data is correct before you can finish your order (CRM clearing).	References UK ICO on keeping personal data up to date

P9 Missing or insufficient Session Expiration	Failure to effectively enforce session termination. May result in collection of additional user-data without the user's consent or awareness.
How to check?	Countermeasures
● Is the logout button easy to find and promoted? ● Is there an automatic session timeout < 1 week (for critical applications < 1 day). ● Are session timeout lengths appropriate to the length required to complete a transaction (long enough) but also to the sensitivity of the data that the session accesses (shorter for higher sensitivity)? ● A single service can support several combinations of session sensitivity and length. Each such available session type should be evaluated.	● Automatic session expiration should be set. Expiration time could differ widely depending on the criticality of the application and data. ● Session timeout should be no longer than a week and much shorter for critical use cases. A best practice for medium criticality (e.g. webmailer, web shop, social network) is one day as default setting. ● Session timeout should be configurable by the user according to his or her needs. ● If a user has not used the logout-button to finish his session the last time, the user should see a reminder message at next login. ● If the user is unable to logout, or the logout does not terminate the session completely, data may continue to be collected (e.g. tracking sites the user visits elsewhere).
Example	References
When a users forgets to logout from web.de (German mail provider) a popup tells the users at next login that logging out is important for security reasons. Facebook does not implement automatic session expiration. The user has to logout manually. In case the user does not actively log out and someone else uses the device he or she can access or manipulate the user's profile. Amazon implements security without logout button by partitioning the content into different sensitivity levels, and tracking the x-main and session-id cookies. Amazon ensures that only the authenticated user can access personal details, but provides personalized content to a returning user without login.	OWASP Session Management Cheat Sheet Carnegie Mellon Guidelines for Data Protection recommends automatic session timeout besides other controls

P10 Insecure Data Transfer	Failure to provide data transfers over encrypted and secured channels which would exclude the possibility of data leakage. Failure to enforce mechanisms limiting the leak surface, e.g. allowing the inference of any user data out of the mechanics of Web application operation.
How to check?	Countermeasures
● What are the policies for protecting data in transit? ● Is data encrypted during transfer? ● Are secure protocols and algorithms used? ● Are privacy-friendly protocols available for transfer? ● Are private protocols enforced where appropriate? (E.g. Login only available over HTTPS, and sensitive records only accessible by TLS or SFTP)	● Always send personal data by secure protocols i.e. not insecure protocol like ordinary email, many instant messaging clients, FTP. ● Configure transfer protocols so they are secure enough for the types of data being transmitted. ● Allow connections using the best available secure protocols, where possible. ● Disallow weak protocols for sensitive information. ● Avoid personal information in the URL, especially if the data transfer is unencrypted. ● Activate privacy in protocols (e.g. Privacy Extensions in IPv6). ● Support TLS/DTLS, do not support SSLv3. ● Use ECDHE and GCM ciphers, do not support static RSA key exchange and CBC-based ciphers.
Example	References
● Configure services to disable broken security protocols such as SSLv3. ● Configure services to enable the latest secure protocols. ● Enforce HTTPS for the entire Web application session, from first visit to login page to completion of logout. ● Disable vulnerable file transfer services such as Telnet and FTP on file servers. Enable secure transfer protocols instead. About the insecurity of current internet technologies and the initiative to build new ones: http://youbroketheinternet.org/	http://security.stackexchange.com/questions/7790/guidance-for-implementors-of-https-only-sites-server-side Jim Manico's presentation at AppSecEU 2015: HTTPS is better than ever before - Now it's your turn Privacy Extensions in IPv6 Background information: IEEE 802 Tutorial about Designing Privacy into Internet Protocols (July 2014)