

# Protecting User Privacy in Logs | Wang Boyuan's Blog

Geek 2019-04-18 👁 1,280 ⌚ Read for 10 minutes

关注

Original link: [wangbaiyuan.cn](http://wangbaiyuan.cn)

<<< TRAE 2.0 SOLO 出道，一键贯通从灵感火花到上线部署的全程协作 >>>

[Geek](#) 4 days ago [Technology](#) | [Grab the couch](#) 163 articles rated 0 times, average score 0.0: [Collapse] **Article**

## Directory

- [First: Determine what is private data](#)
- [1. Decoupling Privacy Fields](#)
- [2. Avoid personal privacy information in URLs](#)
- [3. Object printing overrides toString method](#)
- [4. Masking Privacy Fields When Outputting Structured Logs](#)
- [5. Incorporate log code review into code review](#)
- [6. Personal information leakage testing is included in QA and automated testing](#)
- [7. "Censor" private information before uploading to the log collector](#)
- [8. Configuring monitoring alerts for personal privacy information in the log system](#)
- [Summarize](#)

The 2019 "315" (National Day) Gala, in which AI-driven harassing phone calls were used, highlighted the importance of protecting personal privacy in the information age. This article shares seven best practices for protecting user privacy in logging.

In stark contrast to the Chinese mentality of "willing to trade privacy for convenience," European and American countries have clearly gone further and earlier in protecting personal privacy. Around the time of the GDPR's release in May 2018, privacy protection rapidly became a higher priority. As a programmer developing international products, my daily work was impacted by this. We put aside our business needs (Stories) and focused on GDPR-related security requirements.

In the healthcare and financial industries, access to sensitive customer data is generally strictly restricted. Especially with the enactment of the European General Data Protection Regulation (GDPR), the consequences for companies leaking personal data are extremely severe. While China currently lags behind in terms of both legal and awareness regarding personal privacy protection, many people have experienced the inconvenience of personal information leaks, with the rise in harassing phone calls being the most obvious example. However, the promulgation of the Cybersecurity Law and the growing awareness among netizens indicate that personal information protection is on the right track.

For some projects targeting Europe and the United States, we have taken a series of relevant actions from the highest level of the company, from top to bottom, such as sorting out our infrastructure architecture diagrams, data flow diagrams, API data field analysis, etc., including protecting personal information in logs.

## **The particularity of security issues**

Personal privacy, like other security issues, is a never-ending task. You can't claim your website is absolutely secure; you can only say, "I've checked all currently known security vulnerabilities and implemented appropriate defenses to ensure maximum

security." Or, you can say, "We've implemented some good security practices, like using dynamic passwords and installing anti-attack and SQL injection plugins on Nginx."

Today's web systems are generally equipped with logging systems for recording access requests and analyzing online incidents. For example, open source systems include ELK, and SaaS systems include DataDog and Sumo Logic.

It is often unavoidable to record some user privacy information during the logging process. While it is true that developers' awareness of personal privacy protection is important, sometimes developers do not necessarily want to spy on user information. For example, if some program exceptions are not properly captured, the call stack will often be output. The parameters of certain methods in these call stacks may contain personal privacy information.

While there's no single, permanent way to prevent personal information from appearing in logs, we can minimize this by implementing the following practices and **integrating** them into our daily development workflow. These practices involve **code-level** technical practices, team **process** optimization, and **testing and operations** measures.

## First: Determine what is private data

---

Before we delve into how to prevent personal privacy data from appearing in logs, let's define what *private data* is :

- Personally Identifiable Data (PII): such as Social Security number, data combinations (such as first name + date of birth or last name + zip code) or user-generated data (such as email or username, such as [BillGates@hotmail.com](mailto:BillGates@hotmail.com) ), mobile phone number.
- Health Information
- Financial data (such as credit card numbers)
- password

- IP address: IP addresses can also be personal privacy data, especially when they are tied to personally identifiable data. (The 2019 3.15 Gala introduced a way to turn MAC addresses into PII.)

Personal privacy information is diverse, and its definition may need to be completed in cooperation with security experts familiar with GDPR. Based on the actual situation, a thorough review of the data within the application should be carried out to determine what is sensitive.

## 1. Decoupling Privacy Fields

---

When handling private data, the frequency with which the system uses this data should be minimized. For example, when designing a database table, consider using email addresses, or, in an extreme case, ID numbers (PIDs), as the primary key for the "User" table. This means that whenever the system accesses user data, it must use email addresses or PIDs to establish relationships. This may be convenient, and the system will still work, but it significantly increases the exposure of sensitive fields. The more places they appear, the greater the chance they will be logged.

Therefore, a better approach is to decouple private data and use it only when necessary. A common solution is to use a randomly generated string as the ID of the user table and establish a "one-to-one" database table to store the relationship between the user ID and the primary key of the user database table. For example:

 Experience AI code assistant 

```
1  PID | 外键
2  -----
3  42-12xxxx-345 | 5a2_cXKrt32DcW0JpJlyhr7FhTcLPfvLEAb1eA2Hza
```

All database tables except the user table should use this random ID for querying. Even if this random ID is exposed, it will not leak any personal data.

## 2. Avoid personal privacy information in URLs

---

For example, if you have a RESTful API that searches for user information by email, you might easily have an endpoint like `/user/<email>`. This type of request URL is usually logged by the reverse proxy server and web server, so the email will appear in the log. To prevent sensitive data from appearing in the URL, you can

**Option 1.** Don't use sensitive fields as unique identifiers, use these random IDs instead.

**Option 2.** Pass sensitive values as POST data

Similar to the database decoupling of privacy fields mentioned above, these issues need to be considered early in API or database design, otherwise significant refactoring efforts may be required later. The prerequisite for this is to identify which data in the system is sensitive.

## 3. Object printing overrides toString method

---

To troubleshoot or debug issues, developers often add debugging information to the log. For convenience, they might write something like this (printing User directly instead of user.username):

```
1 logger.info("为用户$ {user}更新电子邮件);
```

In some programming languages, such as Java and Javascript, if you print an object directly, it actually prints the string returned by the toString method. In this way, we can override the toString method of the object to avoid the problem of personal information leakage when printing the object.

 Experience AI code assistant 

```
1 class UserAccount {
2     id: string
3     username: string
4     passwordHash: string
5     firstName: string
6     lastName: string
7
8     ...
9
10    public toString () {
11        return "UserAccount (${this.id})";
12    }
```

If the developer is really "suicide", such as directly printing the fields of the object, there is no way to solve it, for example:

 Experience AI code assistant 

```
1 logger
2     .info("The user's details are: ${user.firstName} ${user.lastName}");
3
4
```

## 4. Masking Privacy Fields When Outputting Structured Logs

---

In order to facilitate the viewing of logs, we often upload logs to the log server in the form of JSON strings, so that we can clearly see the key-value pair structure when viewing the logs.

We can traverse all key-value pairs in the application's log output. If the "key" contains a field like firstName, or the "value" matches Email, then replace the corresponding value with "<MASKED>", for example:

 Experience AI code assistant 

```
1  Blacklist = ["firstName", "lastName"]
2  EmailRegex = r".+@.+";
3  class Logger {
4      log(details: Map<string,string>) {
5          const cleanedDetails = details.map( (key, value) => {
6              if (Blacklist.contains(key) || EmailRegex.match(value)) {
7                  return (key, "<MASKED>");
8              }
9              return (key, value);
10         }
11         console.log(JSON.stringify(cleanedDetails));
12     }
13 }
```

## 5. Incorporate log code review into code review

---

Code review is a crucial part of the development process that helps ensure code quality. For example, during code reviews, bugs, robustness issues, and improvement suggestions are often pointed out. Making logging code a key focus for all code reviewers is not a technical aspect, but rather an improvement to the team's code review process.

If you are using [the Pull Request Template](#) to merge code, you may need to set a checkbox in the template to prompt the reviewer to check it.

## 6. Personal information leakage testing is included in QA and automated testing

---

Although most companies currently do not **include personal privacy leakage testing in the scope of work of testers or QA personnel**, this part of the work not only needs to be done by testers, but can even **be automated**.

For example, in a user registration scenario, a tester can simulate a user entering their name and email address on a web front-end form and then check whether the server log contains this information. This can be automated using end-to-end testing tools such as Selenium and Cypress, which then call the log server's API to search for the presence of this information.

Automated personal privacy leakage testing can also be **incorporated into the CI/CD continuous integration pipeline**.

## 7. "Censor" private information before uploading to the log collector

---

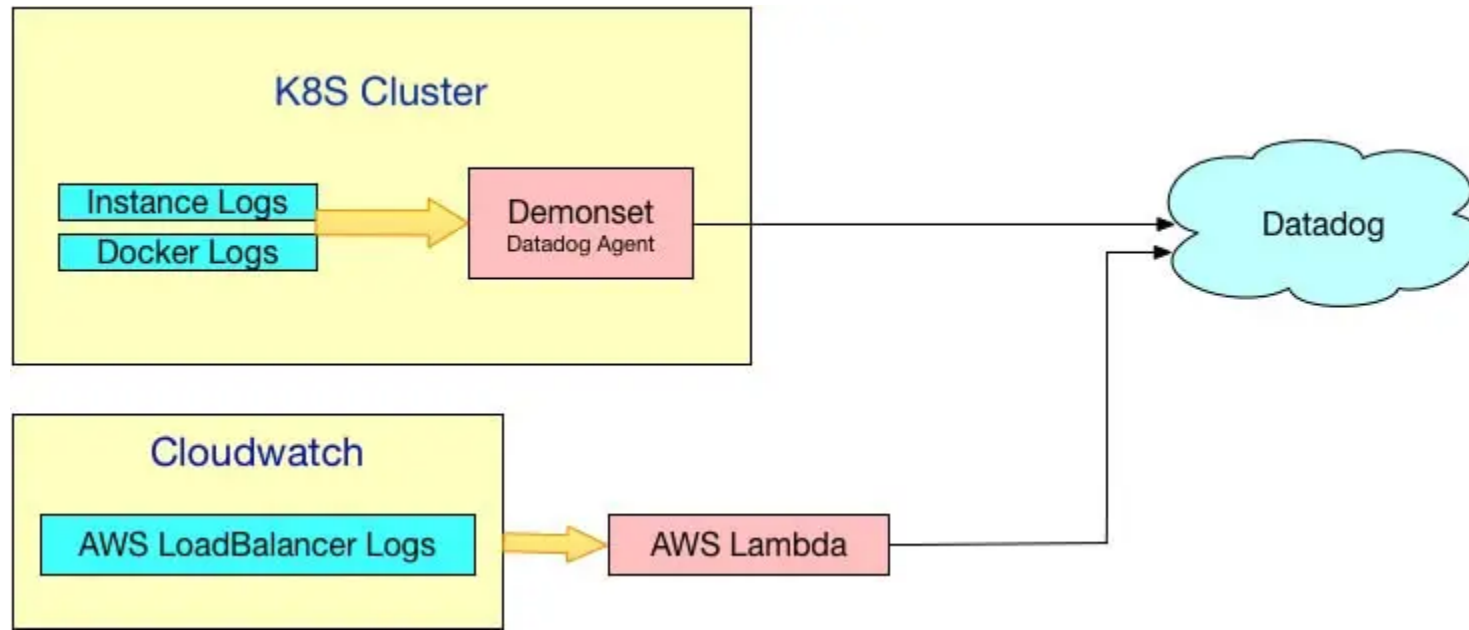
In our project, there are generally two ways to collect logs

- Through the log collection process (agent) provided by the log center, the standard output of the machine instance or the



log file content is pushed to the log server

- Forward logs to the log center through AWS Lambda serverless code




The log collection tool is the only way for logs to reach the log center. By shielding information at this checkpoint, logs from all services (in the case of multiple microservices) can be processed centrally.

## 8. Configuring monitoring alerts for personal privacy information in the log system

Even with the above practices, we still cannot guarantee that personal privacy will never appear in the logs. On the one hand, we can consciously check whether there is any private information when debugging and viewing application logs. On the other hand, we can still **automate this detection work** through some technical means and **notify team members through**

**the alarm system** for processing.

 **[Warn] [Prod][PII] Find some message including email on logs** #env:prod #monitor #pii:email #total  
More than **5** log events matched in the last **10m** against the monitored query: `-@ pii.email.host:(2x OR moz OR demandbase) @ pii.email.host:* - @ pii.email.suffix:(jpg OR png) environment:prod -source:(nginx OR elb OR docker) -@ pii.email.user:ios` by source  
The monitor was last triggered at Thu Apr 18 2019 05:12:46 UTC (**2 secs ago**).

---

[\[Monitor Status\]](#) · [\[Edit Monitor\]](#) · [\[Related Logs\]](#)  
Updated Thu Apr 18 2019 13:12:48 GMT+0800 (China Standard Time) · Created [Thu Apr 18 2019 08:26:48 GMT+0800 \(China Standard Time\)](#) · [Add comment](#) · [Raise priority](#)

▶ 37 events (5 in timeframe)

## Configuring Email Alerts in the Monitoring System

This has already been put into practice in my team. We use Datadog as our logging and monitoring system, and have successfully implemented automatic email notifications when email information appears in our logs. However, it's important to note that while emails are well-matched using regular expressions and are supported by many logging systems, names may have to be handled by artificial intelligence.

## Summarize

## PII Protection

As you can see from the above explanation, protecting personal privacy is no longer a problem that can be solved simply by hiring a security expert, nor is it the responsibility of a single individual. Instead, it requires the collaborative efforts of all roles within the entire team. This is the DevSecOps concept.

- Reference: [medium.com/@joecrobak/...](https://medium.com/@joecrobak/...)

0 people like it 0 people collected and shared it on Weibo [More](#)



[Shares](#) About [Geeks](#): Recording life, engraving the heart; writing, sharing technology! Wang Boyuan's blog is dedicated to exchanging IT experience, and translating and introducing foreign articles to open up the international



perspective of IT. [Author's homepage](#)

- 



[Detailed Explanation of Intent \(Filter\) Usage in Android Development](#)

- 



[How to modify the system time and time zone error in QT](#)



- [【Translation】 Introduction and Installation of Phinx - Phinx Tutorial \(1\)](#)



- [CSS3 rounded border and border shadow example](#)

Previous articleUse [ModSecurity to protect your WordPress blog](#)Next article

Label: Operations and Maintenance

Comments 0



[Login/](#) [Post your comment!](#)

[Register](#)

暂无评论数据

## Table of contents

Close ^

### [The particularity of security issues](#)

First: Determine what is private data

1. Decoupling Privacy Fields
2. Avoid personal privacy information in URLs
3. Object printing overrides toString method
4. Masking Privacy Fields When Outputting Structured Logs
5. Incorporate log code review into code review
6. Personal information leakage testing is included in QA and automated testing
7. "General" protects information before uploading to the log collector

## Search suggestions

Search keywords 

your WordPress [blog](#) Use ModSecurity to protect | [Wang Boyuan's Blog](#)  
of Smart Recommendations [Security and Privacy](#) : How to [Protect User Data](#) and [Privacy](#)  
[Open Source] Terraform enables [within](#) sharing Azure subscriptions [a Team](#) | [Wang Boyuan's Blog](#)  
recommendation systems [in](#) Data privacy issues : [How to protect user privacy](#)  
in uniapp [of user](#) login [data](#) Research on the storage method  
in recommendation systems [Privacy](#) issues : [How to protect user data](#)  
of Image Recognition [Security and Privacy](#) : How to [Protect User Data](#) and [Privacy](#)  
Big [Data](#) and [Privacy Protection](#) : [Early Warning and Response to Data](#) Leakage  
of Language Models [Security and Privacy](#) : [Protecting User Data](#) The Key to  
ClickHouse [data](#) security and [privacy protection](#)

## Featured Content

Say goodbye to manual restrictions: Automate Excel cell data validation with Python

User 83562907... · 31 reads · 0 likes

XxlJob Source Code Analysis 07: Task Execution Process (Part 2) - Trigger Unveiling

Funcy · 21 Reads · 0 likes

Maven build acceleration

Moving · 37 reads · 0 likes

How to implement interface idempotence in Spring Boot project

IT Orange Peel · 58 reads · 1 like

Locate the running Spring Boot program

Emma Song Xia... · 25 Reads · 0 likes

## 为你推荐

### uniapp中用户登录数据的存储方法探究

咕噜企业签名铁蛋   1年前    282    点赞    评论

uni-app

### 推荐系统的隐私问题：如何保护用户数据

OpenChat   1年前    284    点赞    评论

人工智能

### 图像识别的安全与隐私：如何保护用户数据和隐私

OpenChat   1年前    250    点赞    评论


人工智能

### 大数据与隐私保护：数据泄露的预警与应对

OpenChat   1年前    82    点赞    评论

后端   架构   人工智能

### 语言模型的安全与隐私：保护用户数据的关键

OpenChat   1年前    96    点赞    评论

人工智能

### ClickHouse的数据安全与隐私保护

OpenChat   1年前    208    点赞    评论

后端   架构



应用程序中用户隐私合规和数据保护合规的内容模版

咕噜签名分发冰淇淋    1月前    👁 15    👍 点赞    💬 评论    前端

应用程序中用户隐私合规和数据保护合规的内容模版

咕噜分发企业签名梦奇    1年前    👁 81    👍 点赞    💬 评论    iOS

数据交换的数据隐私保护：遵循法规要求和保护用户隐私的方法

OpenChat    1年前    👁 58    👍 点赞    💬 评论    人工智能

数据采集中的隐私保护

用户2527982256116    2年前    👁 207    👍 点赞    💬 评论    前端

Elasticsearch的数据安全与隐私保护

OpenChat    1年前    👁 47    👍 点赞    💬 评论    后端    架构    算法

Elasticsearch的数据安全与隐私保护

OpenChat    1年前    👁 58    👍 点赞    💬 评论    后端    架构    人工智能

将服务开放给用户：构建API接口和用户认证的实践指南 | 青训营

王佳鑫    2年前    👁 90    👍 1    💬 评论    青训营...

Elasticsearch的数据安全和隐私保护

OpenChat    1年前    👁 48    👍 点赞    💬 评论    后端    架构

电商数据合规挑战：API接口在数据隐私保护中的实践

AnthonyI3713612741    9月前    👁 136    👍 点赞    💬 评论    API

