# Google Dorking Report

*This document provides a basic walkthrough of Google Dorking, a technique used to find specific information online using advanced search operators in Google.*

## Exclusive Summary

This report documents the discovery of a high-severity information disclosure vulnerability on a public-facing web server belonging to Stanford University. Through the use of advanced search engine reconnaissance techniques, commonly known as Google Dorking, a publicly accessible Git source code repository was identified. The exposed repository, located at web.stanford.edu/class/cs140/pintos.git,contains the project's complete version control history. This misconfiguration allows any individual to download the full source code, potentially exposing intellectual property, sensitive data within the commit history, and a detailed roadmap of the application's development and security patches. This finding underscores the critical importance of secure deployment practices and regular security audits.

## Introduction & Objective

The primary objective of this investigation was to demonstrate the effectiveness of Google Dorking as a reconnaissance method for identifying security vulnerabilities. By crafting precise search queries, an investigator can compel a search engine to reveal misconfigurations and data exposures that are not readily apparent through normal website navigation. This report serves as a practical case study of this technique, from initial query to vulnerability confirmation.

## Methodology

The investigation was conducted in two distinct stages:

1. **Reconnaissance**: A targeted Google search query was constructed to identify web servers within the stanford.edu domain that were misconfigured to allow directory listing for sensitive .git repositories.
2. **Confirmation & Analysis**: The results of the search query were analyzed, and a promising URL was accessed directly to confirm the presence of the exposed repository and assess the scope of the information disclosure.

## Findings & Analysis

The methodology described above yielded a positive result, confirming the existence of a critical information disclosure vulnerability.

### Reconnaissance via Google Dorking

A specific Google Dork was formulated to query Google's index for the signature of an exposed Git repository.

**Search Query Executed:**

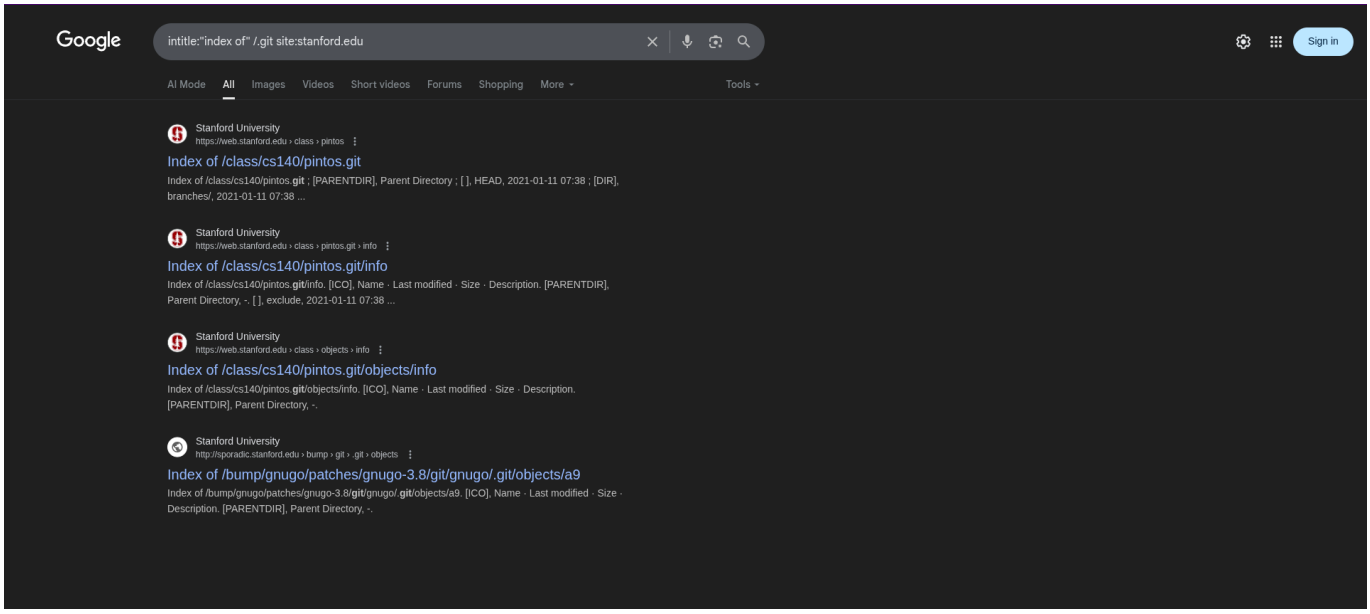
***intitle:"index of" /.git site:stanford.edu***

**Analysis of Query Components:**

- **intitle:"index of":** This operator targets pages with the exact title "index of", which is characteristic of server-generated directory listings.
- **/.git:** This search term looks for the name of the standard Git repository directory within the page's content or URL.

- **site:stanford.edu**: This operator restricts the search exclusively to hosts and subdomains of stanford.edu.

The execution of this query returned a list of URLs that matched these precise criteria, as evidenced by the screenshot below.




## Vulnerability Confirmation

The top result from the reconnaissance phase was investigated further. Upon navigating to the URL ***https://web.stanford.edu/class/cs140/pintos.git,*** the server responded with a full directory listing of the .git repository, confirming the vulnerability.

## Analysis of Exposed Data

The directory listing shown in Figure 2 reveals the entire internal structure of a Git repository. The exposure of these contents, particularly the following directories, is critical:

- **objects/**: The Git object database. This contains all commits, file contents (blobs), and directory structures (trees). Access to this allows for a complete reconstruction of the project's source code and its entire history.
- **refs/**: Contains pointers to branches and tags, revealing the development structure.
- **config**: Contains repository-specific configuration, which can include remote origin URLs and developer email addresses.

## Repository Structure and Contents

The exposed Stanford Git repository contains the Pintos operating system framework, a widely used educational platform designed for the x86 architecture. Pintos serves as an instructional operating system supporting kernel threads, user program execution, and file system operations, making it a cornerstone of computer science education at Stanford and numerous other institutions.

## Educational Context and Institutional Impact

Pintos represents a significant educational asset developed at Stanford University for CS140 (Operating Systems) coursework. The framework has been adopted by over fifty institutions worldwide, including UC Berkeley, Carnegie Mellon, and Johns Hopkins University, establishing it as a critical component of computer science education.

The exposed repository contains not only the current source code but potentially the complete development history, including commit messages, contributor information, and incremental changes made during the course development process. This level of exposure provides unprecedented insight into the educational framework's evolution and implementation details.

## Impact Assessment

1. **Source Code Leakage:** Unauthorized parties can download the complete source code for the "Pintos" project.
2. **Disclosure of Sensitive Information:** Analysis of the commit history (git log) can reveal inadvertently committed secrets, such as API keys, passwords, developer credentials, or comments detailing internal logic.
3. **Intellectual Property Theft:** The code represents intellectual property that is now publicly exposed.

## Conclusion

This investigation successfully demonstrates that Google Dorking remains a potent technique for discovering significant security vulnerabilities stemming from server misconfigurations. The case of the exposed .git repository at Stanford University highlights a critical lapse in secure deployment practices. While the discovery was made for academic demonstration, a malicious actor could easily leverage the same technique for nefarious purposes. It is imperative that organizations implement robust deployment pipelines and conduct regular, proactive security audits to protect their digital assets.