

**January 6, 2010**

## **Visa Data Security Bulletin SQL Injection Attacks**

**Global Payments is pleased to share this information that has been published by Visa.**

To promote the security and integrity of the payment system, Visa is committed to helping financial institutions and payment system participants better understand their responsibilities related to securing cardholder data and protecting the payment industry. As part of the commitment, Visa issues Data Security Bulletins when emerging vulnerabilities are identified in the marketplace, or as a reminder about best practices.

Financial institution clients may share this bulletin with their stakeholders to help ensure they are aware of these emerging vulnerabilities and take steps to mitigate risks.

### **Security Vulnerability**

Recent data security breaches continue to show the prevalence of Structured Query Language (SQL) injection attacks on e-commerce Web sites, corporate Web sites and Web-based applications that manage card accounts (e.g., PIN updates, monetary additions, account holder updates). These attacks also showed how the lack of segmentation between the corporate websites and the payment systems pose serious additional risks to card data stored or transmitted with systems (e.g., Microsoft and UNIX-based) and networks connected to the affected environment.

SQL injection is a technique used to exploit Web-based applications that use client-supplied data in SQL queries. These attacks can occur as a result of improperly designed applications (incorrectly filtered escape characters or error-type handling) residing on un-patched or un-hardened Web and database servers.

### **Recommended Mitigation Strategy**

To minimize the possibility of a SQL injection attack and mitigate the risk of a data compromise, merchants, issuers, acquirers, processors and agents should take the following actions:

- Validate all user input on web-based applications to avoid execution of SQL injection attacks. Sanitize input data by checking for known and expected data

by type, length, format and range. Input validation is used to detect and prevent unauthorized parameters from being passed as a SQL query.

- Do not grant applications DBA or administrative privileges to the database, shared database or shared table. The principle of least privilege should be applied.
- Use only secure Web and database servers. These servers should be hardened to disable default settings and unnecessary services (e.g. ftp, xp\_cmdshell). SQL injection attacks are often further escalated and extended to other systems due to the availability of xp\_cmdshell. Please refer to product vendor Web sites for instructions on hardening Web and database servers (e.g., visit [www.microsoft.com](http://www.microsoft.com) for instructions on hardening IIS Web servers and SQL database servers).
- Use secure applications validated to the Payment Application Data Security Standard (PA-DSS formerly PABP). A list of validated applications is available at [www.pcisecuritystandards.org](http://www.pcisecuritystandards.org).
- Test susceptibility to SQL injection utilizing automated tools and manual techniques.
- Adopt secure coding practices that include regular independent code reviews and testing against SQL injection, for organizations that utilize proprietary or custom applications.
- Update all systems regularly, including Web and database servers, with the current vendor security patches.
- Monitor relevant information security alerts routinely and ensure the latest vulnerabilities from industry resources are addressed (e.g., US-CERT, SANS).
- Purge cardholder data when no longer needed and take steps to ensure prohibited cardholder data (e.g., full magnetic-stripe data, CVV, CVV2, PIN data) is not stored following transaction authorization.

If you have any questions regarding this bulletin, please contact your Relationship Manager.