PAYONE
more than payment.

HINWEISE ZUM BETRIEB SICHERER SYSTEME UND ZUR ENTWICKLUNG SICHERER SOFTWARE

Version 7

Über dieses Dokument

Kontakt

PAYONE GmbH
Fraunhoferstraße 2-4
24118 Kiel
Deutschland

Fon: +49 - 431 25968-0
Fax: +49 - 431 25968-100

www.payone.de
info@payone.de

Markennamen

Informationsschutz

Haftungsausschluss

Änderungsvorbehalt

Inhaltsverzeichnis

1 Gut geschützt im E-Commerce – Sicherheit hat oberste Priorität

Der Schutz von personenbezogenen und vertraulichen Daten sollte selbstverständlich sein – so wie es auch selbstverständlich ist, die Haustür beim Verlassen abzuschließen. Dennoch wird in der Presse immer wieder über Vorfälle von Datendiebstahl berichtet. Dabei kann diesem bereits bei der Entwicklung und dem Betrieb der im E-Commerce genutzten Systeme vorgebeugt werden.

Schützen Sie sich vor Datenlecks und achten Sie bei der Arbeit mit vertraulichen Daten unter anderem auf

- verschlüsselte Übertragungen von sensiblen Daten mit sicheren Protokollen wie z.B. TLS.
- regelmäßige Aktualisierungen Ihrer Systeme, um auch Sicherheitsupdates zu integrieren.
- ihre Firewall-Regeln, denn nur für den Betrieb notwendige Ports sollten erlaubt sein – eingehend sowie ausgehend.
- gesicherten Zugang zu Ihren Datenverarbeitungsanlagen in Form von Passwörtern, biometrischer Identifikation oder Sicherheitsanlagen.

Wir von PAYONE geben Ihnen einige Hinweise, die Ihnen helfen sollen Ihren Shop und Ihre Daten zu schützen.

Bei Fragen, die die spezifischen PAYONE Sicherheitsrichtlinien betreffen, stehen wir Ihnen gerne zur Verfügung.

2 Betrieb sicherer Systeme

2.1 Allgemeine Hinweise

2.1.1 BDSG / Rechtliche Hinweise

Laut §9 BDSG (Bundesdatenschutzgesetz) sind von Stellen, die personenbezogene Daten verarbeiten, erheben oder nutzen, technische und/oder organisatorische Maßnahmen (kurz: TOM) zu treffen, um zu gewährleisten, dass die Sicherheits- und Schutzanforderungen des BDSG erfüllt sind.

2.1.2 Standort (EU-Recht, Safe Harbor)

Bitte beachten Sie die Vorgaben aus dem Bundesdatenschutzgesetz bei der Verarbeitung personenbezogener Daten.

So sollten Ihre Systeme keinen Datentransfer in sogenannte Drittstaaten vornehmen, die über kein dem EU-Recht vergleichbares Datenschutzniveau verfügen. Das bedeutet auch, dass Ihre Server in der EU betrieben werden. Bitte beachten Sie auch aktuelle Diskussionen zum Thema Safe Harbor.

2.2 Wichtige technische Hinweise

2.2.1 Verschlüsselte Datenübertragung

- Personenbezogene Daten sollten stets verschlüsselt übertragen werden (https, aktuelle Verschlüsselungsalgorithmen)
- Unsichere Protokolle wie SSL sollten vermieden werden (Stichwort "POODLE")
- Neben aktuellen Protokollen wie z.B. TLS 1.2 müssen auch die verwendeten Ciphersuiten dem Sicherheitsstandart entsprechen und eine ausreichende Schlüssellänge bieten.
  -> Eine Übersicht über aktuelle Schlüssellängen und sicherer Algorithmen kann z.B. hier http://www.keylength.com/ (http://www.keylength.com/) eingesehen werden.
- Es darf kein Downgrade der Verbindung auf eine unverschlüsselte oder schwach verschlüsselte Verbindung möglich sein
- Es darf kein Downgrade der verwendeten Ciphersuite möglich sein.
- Einen guten Test, um die HTTPS-Fähigkeiten der eigenen Server oder des Browsers zu überprüfen, gibt es hier : https://www.ssllabs.com/ (https://www.ssllabs.com/)

2.2.2 Benutzerkonten und Passwörter

Es gibt heutzutage Zugänge zu vielen, unterschiedlichen Systemen. Der erforderliche Sicherheitslevel hängt davon ab wie kritisch die entsprechenden Systeme und wer die Benutzer dieser Systeme sind. So kann man von Mitarbeitern mit administrativem Zugriff (z.B. auf die PAYONE-PMI, auf die Shop-Administration) eine höhere Sicherheit verlangen als von Endkunden, die sich zu einem Bestellvorgang in einem Webshop anmelden müssen.

- Standard-Zugänge (z.B. root, admin, Administrator, ...) sollten deaktiviert bzw. mit einem hoch-komplexen Passwort versehen werden.
- Jeder Benutzer sollte ein eigenes, persönliches Benutzerkonto erhalten. Generische Benutzer wie "Service", "Buchhaltung", ... sind zu vermeiden.
- Die Zugänge (sowohl reale Benutzer als auch interne, von Programmen verwendete, Benutzer) sollten nur die unbedingt erforderlichen Rechte benötigen.
- Passwörter sollten generell gehasht gespeichert werden - eine Entschlüsselung ist nicht erforderlich, sondern lediglich ein Vergleich, ob das aktuelle Passwort dem gespeicherten entspricht.
- Passwörter sollten komplex sein und aus Groß- und Kleinbuchstaben sowie Ziffern und Sonderzeichen bestehen.
- Passwörter sollten regelmäßig geändert werden müssen - minimal alle 90 Tage.
- Die betroffenen Benutzerzugänge sollten nach Austritt eines Mitarbeiters deaktiviert werden.
- Nach mehrmaliger Falsch-Eingabe sollte ein Zugang entweder automatisch deaktiviert oder für längere Zeit blockiert werden, um sog. "Brute-Force-Angriffe" zu vermeiden.
- Alle Benutzerzugänge sollten regelmäßig auf deren Notwendigkeit hin überprüft werden, nicht mehr genutzte Zugänge sollten deaktiviert werden.
- Benutzer im Hinblick auf Umgang und Definition von Passwörtern schulen, z.B.:
    - Passwörter sollten keine Daten aus dem sozialen Umfeld (Familie, Haustiere, ...) enthalten
    - Passwörter sollten keine Wörter aus einem Wörterbuch sein
    - Passwörter nie weitergeben oder für andere sichtbar aufschreiben

2.2.3 Aufbau, Installation und Konfiguration der Systeme

2.2.3.1 Aufbau / Struktur

- Web-Server und Datenbank-Server sollten nach Möglichkeit auf unterschiedlichen Systemen laufen und sich in unterschiedlichen Netzwerkzonen befinden
  -> Datenbank-Server sollten sich nach Möglichkeit in einer eigenen Zone befinden, die von außen nicht erreichbar ist.

2.2.3.2 Konfiguration

- Auf allen Servern sollten nicht benötigte Dienste / Pakete deinstalliert bzw. deaktiviert werden
- Erzeugte Logiles sind im Inhalt zu prüfen. Es sollte sichergestellt werden, dass so viel wie nötig und so wenig wie möglich geschrieben wird:
    - Reduzieren Sie das Logging sofern möglich auf ein notwendiges Minimum.

- Die Kreditkartennummer **darf nie** komplett geschrieben werden. Es ist maximal eine Ausgabe im Format 6X4 oder X4 erlaubt, z.B. 412345xxxxxx1234 oder besser xxxxxxxxxxxx1234 erlaubt.
  Die Pseudokreditkartennummer darf komplett geschrieben werden - sofern Sie das für erforderlich halten.
- Der CVC-Code **darf nie** geschrieben werden.
- Anmelde-Informationen (Benutzername, Passwort) **sollten nie** komplett geschrieben werden.
- Der Portal-Key sollte zu Ihrer eigenen Sicherheit **nie** in Logfiles geschrieben werden.
- Personenbezogene Daten sollten nur maskiert geschrieben werden - empfohlen ist hier eine Maskierung aller personenbezogenen Daten (Adressdaten, E-Mail, Telefonnummer, Bankverbindung, ...)
  -> je weniger Daten in Ihren Logfiles vorhanden sind, desto weniger gefährlich sind diese.
- Die Logfiles müssen so abgelegt und so gesichert werden, dass ein Zugriff von außen **unter keinen Umständen** möglich ist.

2.2.4 Aktualität der Systeme

- Die **Systeme sollten regelmäßig aktualisiert / gepatcht** werden, um **Sicherheitsupdates** zu implementieren und Sicherheitslücken zu schließen.
  Dazu gehören z.B. (diese Liste ist nicht zwingend vollständig und ist abhängig von Ihrer eingesetzten Umgebung):
  - **Betriebssysteme**
  - **Shop-Systeme - soweit im Einsatz und deren Komponenten / Module / Plugins**
  - **Router, Switches, Firewalls, ggf. Load-Balancer**
  - **ggf. weitere Systeme**
- Virenscanner sollten regelmäßig aktualisiert und deren Logfiles geprüft werden.
- **Beachten Sie aktuelle Nachrichten bzgl. ihrer eingesetzten Systeme in der Presse und im Internet,** z.B.: http://www.heise.de/security/ (http://www.heise.de/security/)

2.2.5 Schutzsysteme

2.2.5.1 Firewalls

- Die Firewalls sollten nur für den Betrieb notwendigen Ports erlauben
  -> alle anderen Ports sollten geschlossen / geschlossen sein.
- Dieses gilt sowohl für eingehende als auch für ausgehende Ports - nur die unbedingt erforderlichen Ports sollten erlaubt sein
- Die Firewall-Regeln sollten regelmäßig auf deren Notwendigkeit überprüft werden.

2.2.5.2 Virenscanner

- Auf allen Systemen (Windows wie Linux) sollten Virenscanner mit entsprechendem Monitoring (Alarmieren von Auffälligkeiten und fehlgeschlagenen Signatur-Updates) installiert sein.

2.2.5.3 File Integrity Monitoring

- Auf allen Systemen sollten Systeme zur Prüfung der Datei-Integrität installiert sein, um Veränderungen auf Dateiebene erkennen zu können.

2.2.5.4 Rootkit-Erkennung

- Die Rootkit-Erkennung zielt auf das Erkennen einer Veränderung spezieller Dateien ab, welche beim Booten des Systems involviert sind.

2.2.6 Hosting / Housing / eigener Betrieb

Im Folgenden werden einige Varianten zum Server-Betrieb beschrieben. Der genaue Umfang ist in den SLAs und Verträgen zwischen Ihnen und Ihrem Dienstleister (auch Hoster genannt) definiert.

2.2.6.1 Webhosting

Beim Webhosting nutzen Sie fertig konfigurierte Server (meist virtualisierte Server) des Hosters, der auch die Aktualisierung der Systeme vornimmt.

Achten Sie bei der Auswahl des Dienstleisters darauf, dass dessen Systeme aktuelle Verschlüsselungsprotokolle inkl. aktueller Ciphers anbieten.

Bitte beachten Sie, dass Sie aber für die Aktualisierung der darauf installierten Applikations-Software (Shop-Systeme, ...) selbst verantwortlich bleiben.

Für Backups werden Ihnen meist entsprechende Services angeboten, die Sie selbst lediglich konfigurieren müssen und entsprechend nutzen können.

2.2.6.2 Managed Server

Bei sog. Managed Servern ist i.A. der Hoster für die Aktualisierung der Server (Betriebssystem und Datenbank) sowie der Infrastruktur verantwortlich. Damit sollte bereits ein großer Teil der Wartung vom Hoster erledigt werden.

Achten Sie bei der Auswahl des Dienstleisters darauf, dass dessen Systeme aktuelle Verschlüsselungsprotokolle inkl. aktueller Ciphers anbieten.

Bitte beachten Sie, dass Sie aber für die Aktualisierung der darauf installierten Applikations-Software (Shop-Systeme, ...) selbst verantwortlich bleiben.

Für Backups werden Ihnen meist entsprechende Services angeboten, die Sie selbst lediglich konfigurieren müssen und entsprechend nutzen können.

2.2.6.3 Root-Server

Bei sog. Root-Servern bekommen Sie vom Hoster Server-Systeme bereitgestellt, welche einmalig mit dem vereinbarten Betriebssystem in der jeweils aktuellen Version ausgestattet sind.

Für die Pflege und Wartung der Systeme sind Sie alleine verantwortlich - es sei denn Sie haben anderslautende SLAs mit dem Hoster vereinbart.

D.h.: Sie müssen die Systeme jeweils aktuell halten und auf Sicherheitslücken überprüfen.

Die Infrastruktur (Firewalls, Switche, ggf. Load-Balancer, ...) werden vom Hoster betreut, überwacht und aktuell gehalten.

Für die Applikation und Backups bleiben Sie selbst verantwortlich.

2.2.6.4 Server Housing (Co-Location)

Beim Server Housing verwenden Sie Ihre eigenen Server, welche beim Dienstleister untergebracht sind.

Sie nutzen ausgewählte Dienste wie Stromversorgung, USV, Klimatisierung, Netzwerk-Infrastruktur des Dienstleisters.

Der genaue Umfang unterliegt einer entsprechenden Definition in den SLA zwischen Ihnen und Ihrem Dienstleister.

2.2.6.5 Eigener Serverbetrieb

Bei dieser Variante sind Sie für alle Bereiche des Serverbetriebes verantwortlich.

2.2.7 Test und Überwachung

Prüfen Sie die Zuverlässigkeit der installierten Maßnahmen sowie deren Funktionsfähigkeit im Falle einer Alarmierung:

- Schwachstellen-Scan / Vulnerability Scan
  Bei einem Schwachstellen-Scan wird die verwendete Software ermittelt und gegen eine Schwachstellendatenbank abgeglichen, hier fallen dann Sicherheitslücken in der verwendeten Software auf, z.B. Buffer Overflow im verwendeten ssh-Server.
  Dieser Scan finden keine Schwachstellen auf der Applikationsebene wie z.B. SQL-Injection oder XSS-Injection. Das Verfahren ist minimal inversiv und sollte das zu testende System kaum beeinträchtigen.
- Penetrations Test
  Weiterführende Ergebnisse werden beispielsweise mit Penetrationstests erreicht. Hier wird versucht, in ein System einzudringen - ähnlich wie es auch ein Hacker tun würde. Der Pentest findet über alle möglichen Ebenen einer Applikationsstruktur statt (Netzwerkebene, Applikationsebene, Benutzerebene). Je nach Ausprägung des Pentests kann dieser deutlich inversiver als der Schwachstellen-Scan sein und das zu testende System teils stark belasten.

2.3 BfDI, TOM

Dieser Abschnitt stellt eine Kopie des Wiki des **Bundesbeauftragte für den Datenschutz und die Informationsfreiheit** (BfDI) dar:
http://www.bfdi.bund.de/bfdi_wiki/index.php/Technische_und_organisatorische_Ma%C3%9Fnahmen
(http://www.bfdi.bund.de/bfdi_wiki/index.php/Technische_und_organisatorische_Ma%C3%9Fnahmen)
.

2.3.1 Abgrenzung technisch und organisatorisch

Unter **technischen Maßnahmen** sind alle Schutzversuche zu verstehen, die im weitesten Sinne physisch umsetzbar sind, wie etwa

- Umzäunung des Geländes
- Sicherung von Türen und Fenstern
- bauliche Maßnahmen allgemein
- Alarmanlagen jeglicher Art

oder Maßnahmen die in Soft- und Hardware umgesetzt werden, wie etwa

- Benutzerkonto
- Passworterzwingung
- Logging (Protokolldateien)
- biometrische Benutzeridentifikation

Als **organisatorische Maßnahmen** sind solche Schutzversuche zu verstehen die durch Handlungsanweisung, Verfahrens- und Vorgehensweisen umgesetzt werden. Beispiele hierfür sind

- Besucheranmeldung
- Arbeitsanweisung zum Umgang mit fehlerhaften Druckerzeugnissen
- Vier-Augen-Prinzip
- festgelegte Intervalle zur Stichprobenprüfungen

2.3.2 TOM gemäß Anlage zu § 9 BDSG

Die Anlage zu gibt vor, in welchen Kategorien Schutzmaßnahmen sichergestellt sein müssen. Nachfolgend werden die einzelnen Anforderungen nebst Beispielen beschrieben.

2.3.2.1 Zutrittskontrolle

Gemeint sind Maßnahmen um zu verhindern, dass Unbefugte Zutritt (räumlich zu verstehen) zu Datenverarbeitungsanlagen erhalten, mit welchen personenbezogene Daten verarbeitet werden.

- Gebäudesicherung
    - Zäune

  - Pforte
  - Videoüberwachung
- Sicherung der Räume
  - Sicherheitsschlösser
  - Chipkartenleser
  - Codeschlösser
  - Sicherheitsverglasung
  - Alarmanlagen

2.3.2.2 Zugangskontrolle

Gemeint sind Maßnahmen um zu verhindern, dass Datenverarbeitungsanlagen von Unbefugten benutzt werden können, wobei allerdings das Wort "nutzen" sich nicht auf die Legaldefinition des § 3 Abs. 5 (http://www.bfdi.bund.de/bfdi_wiki/index.php/3_BDSG) BDSG beschränkt.

- Zugang zu Rechnern/Systemen (Authentifizierung)
  - Benutzerkennung mit Passwort
  - biometrische Benutzeridentifikation
  - Firewall
  - zertifikatsbasierte Zugangsberechtigung

2.3.2.3 Zugriffskontrolle

Es muss gewährleistet werden, dass die zur Benutzung von DV-Anlagen berechtigten Nutzer ausschließlich auf Inhalte zugreifen können für welche sie berechtigt sind **und** das personenbezogene Daten bei der Verarbeitung und Nutzung und nach dem Speichern (http://www.bfdi.bund.de/bfdi_wiki/index.php/Speichern) nicht unbefugt kopiert, verändert oder gelöscht (http://www.bfdi.bund.de/bfdi_wiki/index.php/L%C3%B6schen) werden können.

- Berechtigungskonzept
- Benutzerkennung mit Passwort
- gesicherte Schnittstellen (USB, Firewire, Netzwerk, etc.)
- Datenträgerverwaltung
- zertifikatsbasierte Zugriffsberechtigung

2.3.2.4 Weitergabekontrolle

Es muss verhindert werden, dass personenbezogenen Daten bei der elektronischen Übertragung oder beim Transport oder bei der Speicherung auf Datenträgern unbefugt gelesen, kopiert, verändert oder gelöscht werden können und das festgestellt werden kann an welchen Stellen eine Übermittlung solcher Daten im DV-System vorgesehen ist.

- Sicherung bei der elektronischen Übertragung
    - Verschlüsselung (http://de.wikipedia.org/wiki/Verschl%C3%BCsselung)
    - VPN (http://de.wikipedia.org/wiki/Virtual_Private_Network)
    - Firewall (http://de.wikipedia.org/wiki/Firewall)
    - Fax-Protokoll (http://de.wikipedia.org/wiki/Fax#Fax-Protokoll)

- Sicherung beim Transport
    - Verschlossene Behälter
    - Verschlüsselung

- Sicherung bei der Übermittlung
    - Verfahrensverzeichnis
    - Protokollierungsmaßnahmen

2.3.2.5 Eingabekontrolle

Es muss sichergestellt werden, dass nachträglich überprüft werden kann ob und von wem personenbezogene Daten eingegeben, verändert oder gelöscht worden sind.

- Protokollierung
- Benutzeridentifikation

2.3.2.6 Auftragskontrolle

Es muss sichergestellt werden, dass personenbezogene Daten die im Auftrag verarbeitet (http://www.bfdi.bund.de/bfdi_wiki/index.php/Auftragsdatenverarbeitung) werden, gemäß den Weisungen des Auftraggebers verarbeitet werden.

- Weisungsbefugnisse festlegen
- Vor-Ort Kontrollen
- Datenschutzvertrag gemäß den Vorgaben nach § 11 (http://www.bfdi.bund.de/bfdi_wiki/index.php/11_BDSG) BDSG
- Stichprobenprüfung
- Kontrollrechte

2.3.2.7 Verfügbarkeitskontrolle

Es muss sichergestellt werden, dass personenbezogene Daten gegen zufällige Zerstörung oder Verlust geschützt werden.

- Brandschutzmaßnahmen
- Überspannungsschutz
- Unterbrechungsfreie Stromversorgung
- Klimaanlage
- RAID (Festplattenspiegelung)
- Backupkonzept
- Virenschutzkonzept
- Schutz vor Diebstahl

2.3.2.8 Trennungsgebot

Es ist sicher zu Stellen, dass personenbezogene Daten, die zu unterschiedlichen Zwecken erhoben wurden getrennt verarbeitet werden können.

- Trennung von Produktiv- und Testsystemen
- getrennte Ordnerstrukturen (Auftragsdatenverarbeitung)
- separate Tables innerhalb von Datenbanken
- getrennte Datenbanken

Insbesondere sind allgemein Verschlüsselungsverfahren nach aktuellem Stand der Technik zu berücksichtigen.

2.3.3 Prinzip der Verhältnismäßigkeit

Das BDSG schränkt sich in den zu treffenden Schutzmaßnahmen selbst ein. Für TOM gilt ein sog. Verhältnismäßigkeitsprinzip (http://www.bfdi.bund.de/bfdi_wiki/index.php?title=Verh%C3%A4ltnism%C3%A4%C3%9Figkeit&action=edit&redlink=1). Demnach müssen personenbezogene Daten nicht unendlich stark geschützt werden, wenn die Maßnahmen dafür wirtschaftlich unangemessen hoch ausfallen würden. Daraus lässt sich ableiten dass bei einer Auftragsdatenverarbeitung (ADV) der Dienstleister, welcher nur einen Teil der Daten zur Bearbeitung erhält, nicht zwingend die gleichen Schutzmaßnahmen treffen muss, wie sie etwa die verantwortliche Stelle ausführt.

Beispiel: Der EDV-Dienstleister einer Bank kann (aus wirtschaftlicher Sicht) nicht die gleichen Sicherheitsmaßnahmen gewährleisten wie die Bank selbst. Da er in aller Regel nur auf einen Teilbereich der Daten Zugriff hat (oder zur Verfügung) ist dies gesetzlich auch nicht geboten, selbst wenn die Daten als Sensibel zu betrachten sind (Kontonummern, Kredikartenumsätze).

2.4 Weiterführende Links

Link	Beschreibung
www.bfdi.bund.de/bfdi_wiki/	Insbesondere Hinweise zu technischen und organisatorischen

Link	Beschreibung
(http://www.bfdi.bund.de/bfdi_wiki/)	Maßnahmen: http://www.bfdi.bund.de/bfdi_wiki/index.php/Technische_und_organisatorische_Ma%C3%9Fnahmen (http://www.bfdi.bund.de/bfdi_wiki/index.php/Technische_und_organisatorische_Ma%C3%9Fnahmen)
Wikipedia Schwachstellen-Scans (http://de.wikipedia.org/wiki/Penetrationstest_%28Informatik%29)	Definition und Umfang von Schwachstellen-Scans
Wikipedia Penetrationstest (http://de.wikipedia.org/wiki/Penetrationstest_%28Informatik%29)	Definition und Umfang von Penetrationstest

3 Entwicklung sicherer Systeme

3.1 Allgemeine Hinweise

Dieses Dokument gibt ergänzende Hinweise zu allgemein bekannten Richtlinien sicherer Software-Entwicklung und weist zusätzlich auf Besonderheiten im Zusammenhang mit dem Umgang personenbezogener Daten und der PAYONE Plattform hin.

Ebenso sind beispielsweise Vorgaben aus OWASP (Open Web Application Security Project) zu berücksichtigen.

3.1.1 Eingabevalidierung

Eingaben sollten vor der weiteren Verarbeitung validiert werden. So sollte beispielsweise der Eingabebereich soweit möglich eingeschränkt und nur sinnvolle Zeichen erlaubt werden. So sollten bei der Programmierung u.a. folgende Punkte beachtet werden:

- Datentyp
- Erlaubter Zeichensatz
- Erlaubte Zeichen
- Minimale, maximale Länge
- Minimaler, maximaler Wertebereich
- Verwendung von Grenzbereichen

3.1.1.1 Folgende Beispiele

Eingabe	Erlaubte Zeichen	Prüfung
Kreditkartennummer, Gültigkeitsdatum	Nur Ziffern mit einer Länge von max. 19 Ziffern	Alles außer Ziffern sollte abgelehnt werden
IBAN, BIC	Nur Großbuchstaben, Ziffern mit einer Länge von max. 34 Zeichen	Kleinbuchstaben sollten in Großbuchstaben konvertiert werden, da die PAYONE Plattform ausschließlich Großbuchstaben und Ziffern erlaubt. -> Kleinbuchstaben sowie Leerzeichen werden abgelehnt.
Bankleitzahl, Kontonummer	Abhängig vom Land des Kontoinhabers	Für Deutschland sind lediglich Ziffern erlaubt und dementsprechend sollte alles andere abgelehnt werden. Die PAYONE Plattform errechnet für DE-BLZ/Kontonummer automatisch die korrekte IBAN/BIC. In anderen Ländern sind u.U. auch Buchstaben und Zeichen wie "." oder "-" in der Kontonummer bzw. Bankleitzahl erlaubt.

Eingabe	Erlaubte Zeichen	Prüfung
Adressdaten	Abhängig vom Land des Endkunden	In Adressdaten sind Groß- und Kleinbuchstaben sowie Ziffern und einige Zeichen wie : . , - / + # ( ) & Beachten Sie, dass auch andere Zeichen aus dem UTF-8 Zeichensatz vorkommen könnten, sofern Sie beispielsweise Adressen von Endkunden aus China o.ä. haben.
Dateitransfer	nicht zutreffend für die API	Wenn Sie Dateien übertragen sollten (z.B. Bilder, CSV-Dateien, ...), dann ist eine Prüfung auf den Dateityp (z.B. MIME-Type) sowie die Maximal-Größe sehr zu empfehlen.

3.2 Logfiles

3.2.1 Was darf / sollte / darf nicht in Logfiles geschrieben werden

Logfiles sind zur Fehlersuche und -analyse sehr sinnvoll und notwendig. Achten Sie dabei bitte auf folgende Punkte:

- Die Kreditkartennummer **darf nie** komplett geschrieben werden. Es ist maximal eine Ausgabe im Format 6X4 oder X4 erlaubt, z.B. 412345xxxxxx1234 oder besser xxxxxxxxxxxx1234 erlaubt.
  Die Pseudokreditkartennummer darf komplett geschrieben werden - sofern Sie das für erforderlich halten.
- Der CVC-Code **darf nie** geschrieben werden.
- Anmelde-Informationen (Benutzername, Passwort) **sollten nie** komplett geschrieben werden.
- Der Portal-Key sollte zu Ihrer eigenen Sicherheit **nie** in Logfiles geschrieben werden.
- Personenbezogene Daten sollten nur maskiert geschrieben werden - empfohlen ist hier eine Maskierung aller personenbezogenen Daten (Adressdaten, E-Mail, Telefonnummer, Bankverbindung, ...)
  -> je weniger Daten in Ihren Logfiles vorhanden sind, desto weniger gefährlich sind diese.
- Die Logfiles müssen so abgelegt und so gesichert werden, dass ein Zugriff von außen **unter keinen Umständen** möglich ist.

Bitte bedenken Sie:

- Der API-Request an die PAYONE Plattform sollte **nie** komplett in Logfiles geschrieben werden, da hier beispielsweise der Portal-Key und personenbezogene Daten enthalten sind.

3.2.2 Server-Logs

- (Application- Web-, ...)Server schreiben u.U. Logfiles, in denen Fehlerzustände (sog. Exceptions) geschrieben werden.
  -> Auch hier ist darauf zu achten, dass auch hier der Inhalt nach o.g. Regeln verborgen bzw. maskiert wird, um zu verhindern, dass sensible Daten im Klartext bzw. unmaskiert geschrieben werden.

3.2.3 Allgemein

- Logfiles sollten nie im Zugriffspfad des Web- oder Applicationservers liegen - auch dann nicht, wenn der Zugriff durch Techniken wie ".htaccess" gesichert ist.
- Logfiles sollten nur mit minimalen Zugriffsrechten für einen möglichst kleinen Anwenderkreis geschrieben werden.
- Auch automatisch generierte Mails, die beispielsweise über Fehler informieren sollen, dürfen keine vertraulichen Daten enthalten.
- Je weniger Sie in Logfiles speichern desto besser ist es falls diese doch in unberechtigte Hände fallen (Datensparsamkeit).
  Bedenken Sie beispielsweise, dass ein Server-Administrator stets Zugriff auf die Logfiles hat - je weniger Daten er selbst sehen kann, desto sicherer sind diese Daten.

3.3 Authentifizierung und Sessions

- Verwenden und erzwingen Sie komplexe Passwörter. Dazu gehören eine Kombination aus Groß- und Kleinbuchstaben sowie Ziffern und Sonderzeichen sowie eine erzwungene Mindestlänge. Das gilt auch für automatisch erzeugte Passwörter.
- Die regelmäßige Änderung von Passwörtern sollte erzwungen und Standardpasswörter dürfen nicht verwendet werden.
- Vermeiden Sie konkrete Hinweise darauf, ob der Benutzername oder das Passwort ungültig oder abgelaufen oder eventuell auch gar nicht vorhanden sind. Zeigen Sie lediglich an, dass eine Anmeldung mir der Kombination aus Benutzerkennung in Passwort nicht möglich ist.
- Speichern Sie keine Passwörter im Klartext, sondern lediglich als Hashwert (Methoden wie MD5 oder SHA-1 sind hierzu nicht mehr zulässig).
- Verwenden Sie stets verschlüsselte Verbindungen (https) zur Übertragung vertraulicher Daten (inklusive der Anmeldedaten und auch der Session-Id). SSL Verschlüsselung ("POODLE") ist veraltet und nicht mehr ausreichend. Verwenden Sie statt SSL den noch als sicher geltenden TLS Standard in der jeweils aktuellen Version.
- Benutzersitzungen (Sessions) müssen nach einer bestimmten Zeit der Inaktivität automatisch ablaufen.
- SessionIds (o.ä.) sollten nicht in einer URL übergeben werden, da diese Informationen so leicht einsehbar und verfälschbar sind.
- Wenn Cookies zum Speichern von Sessiondaten verwendet werden, so sollten diese Cookies mit dem "http-Only"- und dem "Secure"-Flag versehen sein.

3.4 Programmierung (OWASP)

Es sind allgemeine Regeln (OWASP) zu befolgen, u.a. in Bezug auf:

- Broken Authentification und Session-Management
- SQL Injection

- Cross Site Scripting (XSS)
- Buffer Overflows
- Injection Flaws
- Information Leakage and Improper Error Handling
- Verschlüsseltes Speichern von Daten
- Security Misconfiguration
- Insecure Remote File Includes
- Insecure Direct Object Reference
- Insecure Transport Layer Protection
- Failure to Restrict URL Access
- Cross Site Request Forgery (CSRF)
- Unvalidated Redirects and Forwards
- Insecure Cryptographic Storage
- Insecure Transport Layer Protection

Außerdem:

- Exception-Handling, z.B. keine Stacktraces anzeigen - aus denen kann ein Angreifer gezielt Informationen auf Ihr System entnehmen
- Dependency-Management für verwendete Frameworks
  -> wenn Sie Frameworks verwenden, dann
    - achten Sie darauf, dass diese aktuell sind und auch in der Zukunft gepflegt werden
    - prüfen Sie aktuelle Nachrichten zu verwendeten Frameworks auf Informationen zu Schwachstellen ("vulnerability") und Sicherheitslücken
    - verwenden Sie nach Möglichkeit stets die aktuellste Version des Frameworks

3.5 Tests

- Führen Sie ausführliche Tests mit Ihrer entwickelten Software und der eingesetzten Werkzeuge durch.
- Diese Tests sollten nach Möglichkeit automatisiert sein, um auch bei fortlaufender Software-Pflege stets die Funktionsfähigkeit bereits getesteter und unveränderter Software-Teile gewährleisten zu können.
- Berücksichtigen Sie bei den Tests stets nicht nur den "Gut-Fall", sondern auch:
    - den "Schlecht-Fall" (Eingabe außerhalb des erlaubten Wertebereiches, ungültige Werte, ...) und
    - den "Grenz-Fall" (Eingabe mit erlaubtem Min-/Max-Wert, z.B. Betrag = "0", Betrag = <leer>).

3.6 Weiterführende Links

Link	Beschreibung
https://www.owasp.org (https://www.owasp.org)	Link zur Open Web Application Security Project (OWASP, einer Non-Profit-Organisation mit dem Ziel, die Sicherheit von Anwendungen und Diensten im World Wide Web (http://de.wikipedia.org/wiki/World_Wide_Web) zu verbessern.
https://www.owasp.org/index.php/OWASP_Broken_Web_Applications_Project (https://www.owasp.org/index.php/OWASP_Broken_Web_Applications_Project)	Beispiele wie es nicht gemacht werden sollte.