


# X.25 WITHIN THE PAYMENT CARDS INDUSTRY

## ADVANCING COMMERCE™



The *Payment Card Industry Data Security Standard* (PCI DSS) states that any system that stores, transmits, or processes cardholder data, or any system connected to such a system, is within the scope of the cardholder data environment (CDE) and must be protected with PCI DSS controls. Unfortunately, X.25 networks are sometimes overlooked in the PCI DSS scoping process, as they are often part of legacy networks or are assumed to be exclusive, private networks and thus sometimes not top-of-mind for the reviewer.

There are two primary types of X.25 networks: those that use dedicated lines and those that use a carrier provided, switched network. For networks with dedicated lines, a level of control exists inherently through the dedicated connection, in effect making it a private network. In a switched environment, there is the potential for X.25 networks to connect to each other simply by knowing the Data Network Identification Code (DNIC), which is similar to a phone number. This ability of X.25 networks to connect to each other may cause a switched network to be considered a “public” network and therefore need PCI DSS controls. In other words, without proper access controls on a switched network, it may be possible for an unknown X.25 network to make a connection to a merchant’s or processor’s trusted X.25 network.

Attackers are taking advantage of unsecured X.25 networks used by merchants and service providers to steal and compromise their cardholder data. As with any network technology, proper configuration and maintenance of X.25 networks are required to ensure that these networks do not pose a risk to the safeguarding of cardholder data. This whitepaper provides a high-level, technical overview that can be used as guidance by an assessor during a PCI DSS compliance review, or for the owner of the network to better understand how to implement security controls that satisfy PCI DSS requirements.

The technical guidance in this document is provided by Foregenix, a Qualified Security Assessor (QSA), Payment Application QSA (PA-QSA), and a PCI Forensic Investigator (PFI) with experience in X.25 networks.

*This document provides guidance only and should be used as an informational supplement when evaluating an entity’s PCI DSS compliance. As with any technical guidance, the suggestions in this document should be carefully evaluated along with other factors that apply to an entity’s unique environment. The information set forth in this document does not guarantee PCI DSS compliance and does not change or override any PCI DSS requirement.*

BY JOSHUA KNOPP, CISSP MASTERCARD WORLDWIDE



For more information on MasterCard PCI 360 education resources, please visit [www.mastercard.com/pci360](http://www.mastercard.com/pci360) or [www.mastercard.com/sdp](http://www.mastercard.com/sdp).

©2012 MasterCard. Proprietary and Confidential. All rights reserved.  
Advancing Commerce is a trademark of MasterCard International Incorporated.



FORGENIX

Digital Forensics & Incident Response

Foregenix Ltd  
Wesley House, Bull Hill  
Leatherhead, Surrey  
United Kingdom  
+44 (0)845 309 6232

X.25 Within the Payment Card Industry

January 2010

### ***Abstract***

*This paper provides a high level overview of the X.25 protocol suite and some of the security controls available within that environment as they generally relate to financial service organisations and the Payment Card Industry specifically. Due to the number of financial institutions that are presently reliant on X.25 connectivity for at least a portion of their operations, it was felt that specific awareness or even inclusion of the technologies within the Payment Card Industry Data Security Standard (PCI DSS) would improve the overall security posture of the industry and generally reduce the risk of further Account Data Compromises relating to X.25 enabled systems.*

*Foregenix is an independent security consultancy and Qualified Forensic Investigator operating within the Payment Card Industry. Foregenix does not consider itself a subject matter expert in the field of X.25 however and presents this information in the understanding that the information is valid and accurate based largely on its own experiences with X.25 systems as well as in the investigation of Account Data Compromises relating to X.25 environments. Foregenix offers no guarantee or warranty that the information enclosed will thwart or address system attacks or breaches and accepts no liability in the event that parties experience loss or damage of any description after implementing controls discussed in this paper.*

## 1 Overview

X.25 is a collective term for a suite of protocols that was developed in the 1960's for Wide Area Network (WAN) connectivity over Packet Switched Networks. X.25 became popular for implementing data communications in the 1970's and 1980's as it offered technical and commercial advantages over the existing solutions. The technology was adopted by many large corporate users, including major banks and other financial institutions, military and government as well as educational institutes.

An implementation of the technology comprises three distinct components, namely the Packet Switching Exchange (PSE), the Data Circuit-terminating Equipment (DCE) and the Data Terminal Equipment (DTE). These are described briefly below and graphically in Illustration 1 which depicts their relationship with the other devices.

- **Packet Switching Exchange (PSE)** – These are the devices which comprise what is generally known as the *Cloud* and are part and parcel of the carrier or solution provider's infrastructure. The PSE devices are often interconnected with those of additional carriers or service providers, offering wider coverage and during the late 1980's and 1990's a significant global coverage of X.25 networks was available.
- **Data Circuit-terminating Equipment (DCE)** – These devices are analogous to modems or switches and connect the terminal equipment—described below—to the Packet Switching Exchange referenced above.
- **Data Terminal Equipment (DTE)** – DTE devices constitute the end points of an X.25 network or circuit and are normally computers or terminals although routers are also common as the adoption of X.25 reduces with time.

Connectivity in X.25 networking is based on sessions or more accurately, circuits which are initiated by a DTE making a *call*. Two types of circuit are used, switched and permanent. As the names imply the Permanent Virtual Circuit (PVC) provides permanently established connections and as such do not require the DTE devices to initiate a session through a call. The Switched Virtual Circuit (SVC) which is more common however, requires a DTE device to initiate, establish and terminate each session on demand. In order to establish an SVC session the calling or initiating DTE will attempt to CALL the destination DTE by providing the destination address in X.121 format. This address is also known as the Network User Address or NUA.

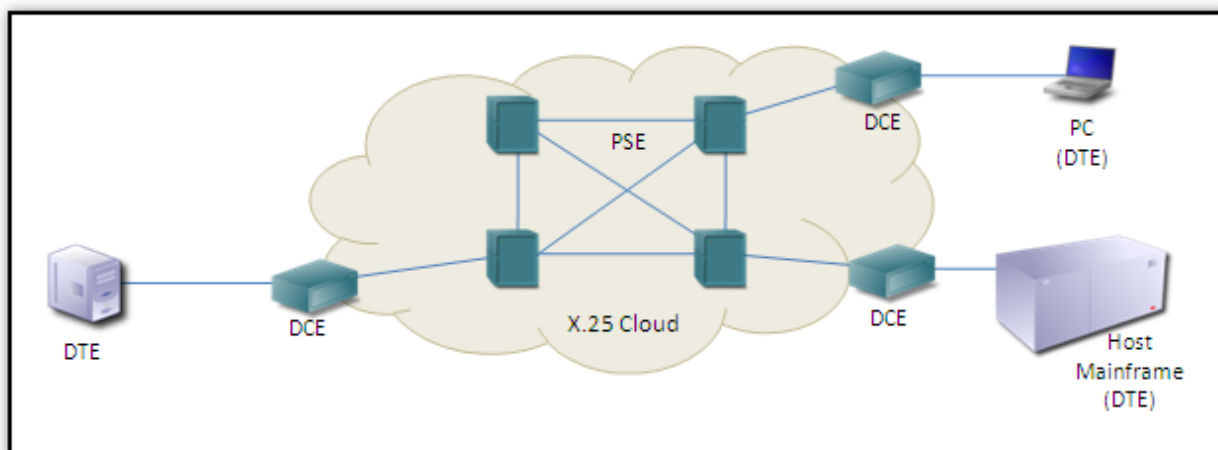


Illustration 1: X.25 Network Example

The X.121 address consists of a number of significant fields that assist in the routing of X.25 data. The first four digits of the address specify the Data Network Identification Code (DNIC) while the remainder of the address represents the National Terminal Number. The DNIC field which is detailed in the International Telecommunication Union (ITU) document *List of Data Network Identification Codes*<sup>1</sup>—comprises a Data Country Code, which as expected defines the country to which the network belongs, and a single digit that specifies the Packet Switched Network (PSN) or service provider. Illustration 2 below provides a simplistic graphic overview of the address structure. It can be likened to a

<sup>1</sup> <http://www.itu.int/ITU-T/inr/forms/files/dnic-1508-en.pdf>

phone number with an international prefix and city code, which provides international direct dialling.

It is possible to implement a degree of security or access control with the X.25 protocol beyond Permanent Virtual Circuits, and that generally involves Closed User Groups (CUG). The CUG is a Binary Coded Decimal (BCD)<sup>2</sup> value that permits a DTE device to only accept circuit initiation from other DTE's belonging to the same CUG. Essentially a simplistic access control mechanism based on this embedded value.

Two differing types of Closed User Group implementations are supported by the protocol, *basic* and *extended*; however, the extended format may not be available from all service providers. The basic form of CUG supports a parameter length of a single octet to the Closed User Group Selection command. This defines the range of CUG values to be 1 to 99<sup>3</sup>. The extended version supports two octets as the command argument permitting the range 1 through 9999. These along with the related Bilateral Closed User Group functionality are discussed in the following section with the additional security controls.

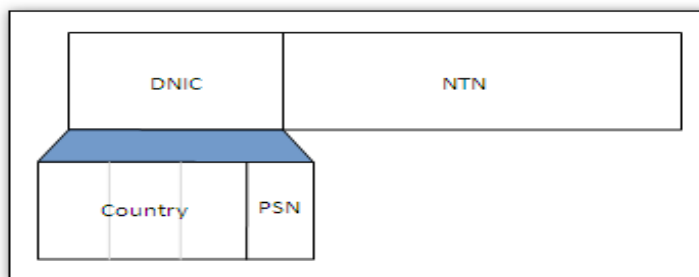


Illustration 2: X.121 Address Structure

In order to achieve an interactive terminal session over the X.25 protocol or connect other asynchronous devices or services, an additional layer of processing and functionality is required. This is provided by the associated protocols X.28, X.29 and X.3 which add packet assembly and disassembly as well as buffering capabilities and is known as a PAD.

In addition to the components mentioned above, users may access the PAD through connecting to an Adapter/Concentrator of Packets (ACP) which transforms the packet protocol from X.25 to an X.28 asynchronous connection that is compatible with the standard modems in use today (or in the very recent past). In this way the user becomes like a DTE, she connects to an ACP and can operate with complete transparency. Generally speaking the user can login to a PAD in either of two ways:

1. **DIRECTLY:** This is through a dedicated wire connection which obviously demands a higher cost, but guarantees a much higher transmission quality.
2. **SWITCHED:** This would interface through a standard telephone line and offers a significantly reduced cost. The transmission quality is however linked to this reduction in cost and in earlier years offered a less than ideal service on occasion.

This direct X.28 user would have her own Network User Address and while some users may have only a single NUA, others can have a multiplexed system offering a number of addresses. This multiplexed system generally consists of one NUA and a variable number of sub-addresses; the actual number of sub-addresses depends on the system configuration as well as the number of interfaces to the PAD. Organisation's often implement these sub-addresses to enable and route certain services within an X.25 enabled host while only a single physical connection is required. The switched user however can only call other DTE, and cannot receive calls or sessions due to not actually having a Network User Address. This lack of NUA can have important security implications.

On typical banking systems such as *IBM AIX* devices (relatively common in X.25 enabled financial organisations), this PAD functionality is divided into a client application—`xspad` for initiating outbound sessions—and a server module that accepts inbound connections. This server module is the X.29 daemon, generally located at `/usr/lib/drivers/pse/x29d` on *AIX* systems, and is a necessary component in accepting inbound terminal sessions over X.25. Similar functionality is available within *HPUX* (another common X.25 enabled technology within financial organisations) where the `padem` application provides the client portion and `/etc/x29server` is the server module.

Within an X.25 environment, configuration information is required by the system providing the end point services (DTE) to address the routing of traffic received. One aspect of the routing mechanism is NUA sub-address mentioned previously while another is the Caller User Data (CUD) field which is used as a protocol demultiplexing identifier. The

<sup>2</sup> [http://en.wikipedia.org/wiki/Binary-coded\\_decimal](http://en.wikipedia.org/wiki/Binary-coded_decimal)

<sup>3</sup> Each nibble of the octet would hold the value 1001 representing 9, producing the maximum CUG value of 99.

hexadecimal value of CC (decimal value 204) denotes the Internet Protocol or IP and should be reviewed carefully if found configured. This is easily achieved on AIX systems with the `xroute` command but other operating systems will obviously differ.

Although many organisations began to adopt Internet based solutions and services in the 1990's many were unable or simply unwilling to migrate solutions over to this new technology. The result being that the now legacy solutions remain intrinsic to the organisations operations but also remain connected to the legacy X.25 infrastructure. As the focus and investment expenditure, including the security aspects, focuses on Internet connectivity few organisations maintain their X.25 systems to the same level. This oversight can and certainly has resulted in a significant number of compromises. In addition to the possible lack of active maintenance, until recently X.25 was considered a private network and as such was beyond the scope of the Payment Card Industry Data Security Standard.

## 2 Security Controls

The X.25 protocol suite as well as the majority of its numerous Operating System implementations has the security features to configure, manage and monitor the X.25 systems, specifically to:

- Protect it from unauthorised incoming calls
- Prevent unauthorised outgoing calls

X.25 security can therefore be used to control access to and from the interconnected networks used by an organisation. The correct use of X.25 security controls can provide an administrator with the facilities to define which remote DTEs are permitted to access the system and which applications within the system are permitted for use. Additionally, it is also possible to control which users on a given X.25 enabled system are permitted to make outgoing calls and which remote DTEs they can call. This does however assume that the standard system access controls are effective; if a user were to achieve privileged `root` access these controls would be largely ineffectual.

The means of implementing these controls unfortunately vary from system to system however, and many modern operating systems no longer natively support the X.25 protocols or infrastructure. This often leaves the configuration and control of the technology in the hands of third party solutions which may be less integrated than native operating system controls.

In order to assess the security of an organisation with X.25 connectivity from the inside, it is important to carefully review the controls implemented and in these circumstances it will almost certainly require the consultant to receive assistance from the network, system or security administrator. This is due to the diversity of technologies in the field and the general lack of knowledge regarding these systems. Once as much information of the controls (including samples) has been collated additional research is often required to assess and understand the information. The following sections present the most common X.25 security control measures.

### 2.1 Access Control Lists (ACLs)

Access controls may be implemented to address both inbound and outbound sessions and are generally applied to the DTE addresses. The implementation of ACLs varies significantly between Operating Systems and implementations. It is important to understand if the ACLs are implemented as white lists or black lists as with any other ACL implementation. As mentioned previously, certain X.25 accesses may result in a client not having an NUA which would obviously influence the application of access controls. Similarly, it is important to ensure a 'catch all' rule is implemented that will address any logical condition not addressed by previous rules.

It is also important to consider the actual termination point of the X.25 interface. Router based end points are generally easier to review as all the configuration information is easily presented as the device configuration. Host based systems may include several configuration and service start-up files. In these instances several attempts may be required to understand the environment and the security configurations with hours of research in between.

## 2.2 Closed User Groups (CUGs)

As mentioned previously, the Closed User Group (CUG) definition is a Binary Coded Decimal (BCD)<sup>4</sup> value that permits a DTE device to only accept communications from other DTE's belonging to the same CUG. This simplistic access control mechanism depends on the embedded value which is controlled by the DTE. The number of available values differs based on the type of CUG available, *basic* and *extended*, (although the extended format may not be available in some circumstances) and provide values 1 through 99 and 1 through 9999 respectively.

As the CUG value is implemented by the DTE the possibility exists that a would be attacker with control over their local DTE (either a host solution or a router device) may be in a position to alter this information at will. This would enable them to scan the desired target systems by trying each available value until success is achieved or the available values are exhausted. This type of attack however depends largely on the environment and the connectivity provided by the organisations service provider as they may implement access controls higher up in the cloud.

A number of X.25 scanners have been written over the years and recent versions have been able to take advantage of modern hardware including multiple threading capabilities, resulting in them being able to scan entire service providers address space in a few hours. Certain PAD client software is also reported to permit a user to provide the CUG value on the command line along with the target NUA although the author has not personally seen such. This would enable the simple bypass of the CUG access controls and would also lend itself to automated scanning of the target.

## 2.3 Bilateral Closed User Groups (BCUGs)

Bilateral Closed User Groups provide a similar function to the standard Closed User Group described previously. Unlike the standard Closed User Group however, this configuration option is designed for a single remote NUA and as such may not be appropriate for many configurations.

## 2.4 Service Availability

Interactive access to an X.25 enabled host is almost certainly a requirement, but interactive access to the host over the X.25 infrastructure is seldom required. As mentioned briefly earlier, in order to achieve interactive communications with a system over X.25, additional protocols and services that understand these protocols are required. Examples of these protocols include X.3, X.29 and X.28.

In order to ensure that an X.25 enabled system presents a robust security stance to the X.25 network it is generally possible to maintain the required data transfer functionality of the host while disabling the interactive login capability. This is described as *generally* as Foregenix is certainly not aware of all X.25 enabled systems and their specific requirements or configuration options. The direct experience of Foregenix in this regard confirms that this approach is possible and effective on those systems at least.

By disabling unused network services on the host system (as required by the PCI DSS) an administrator is able to ensure that login capabilities are disabled for certain protocols while maintaining them for others. For example, by disabling `x29server` on an *HPUX* system or `x29d` on an *IBM AIX* system (through modification of the `inetd` configuration file) login capabilities are disabled over the X.25 network while the Secure Shell or Telnet login capabilities over the local Internet Protocol (IP) network are unaffected.

## 2.5 Effective Logging

An effective control in the security management of a system if not in the actual security implementation is appropriate logging. Systems that support successful and unsuccessful event logging should have this functionality enabled and equally importantly; reviewed on a regular basis. System auditing—if supported—can be invaluable in post security event situations but it's implementation may impact system responsiveness. Additionally, management and effective review of the log data produced can be a challenge.

Provided system event logs, especially successful and unsuccessful login attempts are correctly recorded—if possible centrally—and regularly reviewed it would be a relatively trivial exercise to determine in system attacks are being

---

<sup>4</sup> [http://en.wikipedia.org/wiki/Binary-coded\\_decimal](http://en.wikipedia.org/wiki/Binary-coded_decimal)



attempted over the X.25 infrastructure. Equally so, to determine if user accounts are being accessed over the X.25 networks. This relatively simple mechanism could and would provide early notification of attack, breach and misconfiguration of access control lists.

### 3 X.25 Attack Types

The variation of X.25 attacks and compromises is no where near as diverse as the vulnerabilities that affect TCP/IP (Internet) based connectivity or services. Having said that, at least one well know Solaris exploit that was originally designed to IP based Telnet services was re-engineered to successfully function over X.25 networks although it was far from a trivial exercise taking approximately six months to perfect. Initial system access for system compromise over X.25 networks normally involves the target's use of weak, default or easily guessable user names and passwords and is often termed *Old School hacking* (password guessing).

For this reason (amongst others) it is imperative that PCI DSS and security best practice procedures are implemented with specific regard to user accounts and passwords. Ensuring reasonably complex passwords, requiring regular changing of these passwords and managing dormant and test accounts appropriately are all vitally important. As mentioned previously, interactive access to a host over the X.25 infrastructure is seldom required, applying the PCI DSS principals of removing non-essential network services can disable this functionality for Wide Area Networks. Additional ingress access controls to limit the remote DTE's that may connect to the host provide defence in depth and ensure that a misconfiguration of the available services would unexpectedly present attackers with door on which to focus.

Within certain environments the router based attack that was brought into prominence by *Redkommie* when his video was released as part of the *remote-exploit* video tutorials<sup>5</sup> may be possible. The attack is dependant on a number of factors however, including the routers capabilities and the target organisations additional networking connectivities. This type of attack can take multiple forms depending on the target topology. One possibility involves an attacker creating a loop in the network which could allow them to tunnel the X.25 traffic (encapsulated) to a remote device under their control where they would be able to copy the information before sending it back. This attack follows similar lines to that shown in the video from *Redkommie* mentioned earlier and would enable the attacker to harvest data passing through the X.25 router without actually needing to be logged on.

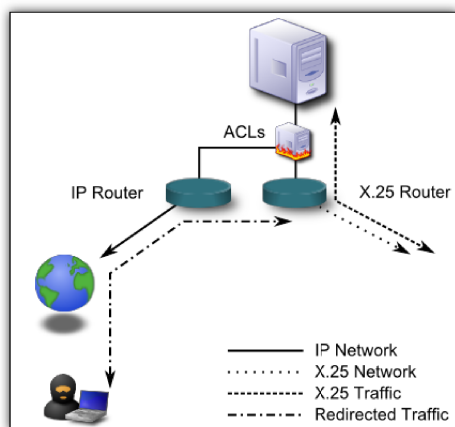


Illustration 3: Possible X.25 Router Hack

An additional option may include a more simple attack with the attacker gaining access to the X.25 router and simply enabling the trace or debug functionality to display dumps of the traffic to the attackers terminal. Provided the attacker enabled session logging on their local machine, they would record all the X.25 data passing through the router.. This attack does require the attacker to remain logged on however, increasing the potential of detection. Additionally, enabling debugging output to the terminal will likely apply significant load on the attacked router and would probably impact performance.

Attacks like these are well suited to the extraction of transaction data and Track data specifically due to its storage format within the industry standard ISO8583 messages. These messages are often transmitted over X.25 circuits between various financial institutions and the track data (stored as a binary encoded decimal) requires no further processing to be extracted from a hexadecimal dump of the traffic. Even if the X.25 traffic did not include ISO8583 messages, data extraction would be a completely trivial exercise for these attackers.

Sniffing attacks can also be possible if an attacker were able to achieve access to an X.25 enabled host. Most operating systems that support X.25 natively at least provide network diagnostic utilities that can be used to monitor and store live X.25 network traffic. For example Foregenix has experience with the *IBM AIX* command `x25mon` being used to store live X.25 network information in an account data compromise. During this incident the compromised network had been breached—resulting in the attacker or attackers having pretty much unfettered access to the entire network—some years previously and had gone largely unnoticed.

In these circumstances it is important to understand what access controls are implemented on the diagnostic tools, and

<sup>5</sup> [http://www.hackerscenter.com/images/videos/126\\_Sniffing\\_Remote\\_Router\\_Traffic\\_via\\_GRE\\_Tunnels\\_%28Hi-Res%29.avi](http://www.hackerscenter.com/images/videos/126_Sniffing_Remote_Router_Traffic_via_GRE_Tunnels_%28Hi-Res%29.avi)



which user accounts may utilise this software. Detection of the tools being actively used or the presence of large trace output files from these tools may indicate a security incident but provided the detected files are still in their raw format, such information is far from conclusive. Detection of parsed data, where the Track data has been extracted and the additional information discarded would and should immediately be cause for concern and in this situation the host and supporting systems should be investigated immediately.

Complete removal of these diagnostic utilities is a possible option to securing an environment; however, if an attacker were successful in obtaining interactive access to the X.25 enabled host there would (eventually) be little in stopping them copying their own version of the required software to the system. Carefully executed sniffing attacks may be difficult to detect, especially those implemented on routing devices as these systems are seldom managed directly unless there are problems to diagnose. Although router configuration is generally rather static (resulting in a reduced requirement for manual maintenance access) it is a sensible option to compare the active, running configuration with the latest approved and locally secured configuration template on a relatively random but regular basis.

## 4 Conclusion

To conclude, although X.25 is a technology from the 1960's, it still holds an important position within many financial organisations. Unfortunately, possibly due to this legacy status, this technology is often overlooked and through this lessened awareness it can pose a significant security risk to organisations.

If treated with the same security scepticism as an Internet facing host, an X.25 enabled system can be appropriately secured. The processes of reducing service availability, managing user accounts and controlling accessibility all tie into security best practice and the PCI DSS in exactly the same way. As with any technology however, the configuration requirements to fulfil these controls differ from system to system and may have their own specific nuances.

Provided the system or host administrators are familiar with the technologies under their responsibility, maintaining a robust and affective security stance through the adoption of the PCI DSS and security best practice should not present a challenge. In the same way, an assessor wishing to review the security controls of an X.25 environment should be able to apply their understanding of both PCI DSS and security best practice to the task. With the assistance of the local administrator to 'interpret' the controls and requirements to the specific technologies local to the environment, the assessor should be readily positioned to form an opinion of the security stance of the environment.