

# Why You Should Use FPGAs in Data Security

Xilinx is an ideal platform for data security applications.

by Mike Nelson

Senior Manager, Storage and Servers, Vertical Markets  
Xilinx, Inc.

[mike.nelson@xilinx.com](mailto:mike.nelson@xilinx.com)

Data security is fast becoming a requirement in communications and enterprise infrastructures. Secure electronic commerce is almost doubling every year. New regulations are mandating the retention and protection of ever more information (Sarbanes-Oxley, HIPAA). Legal liability is dramatically escalating for those who manage such data carelessly. And finally, the value of data as a corporate asset itself is growing in such forms as fully electronic product designs, customer databases, and supply chain management systems. All of these trends make data security a mandatory element in almost any new system architecture.

But the implementation of data security faces a number of serious challenges:

- Performance requirements vary widely
- System cost pressures remain high
- Standards vary widely and are continuously evolving
- Management becomes an integral aspect of the data security landscape as it evolves into a part of the managed IT infrastructure

In this article, I'll review the implementation options for data security and illustrate why Xilinx® FPGAs are a superior platform with which to address this application.

## Implementing Data Security

The implementation of data security can range from pure software to pure hardware. Typically, pure software is an attractive approach, but because of the computational intensity of authentication and encryption algorithms, this approach is inherently limited to single user/client-side applications or very low bandwidth server applications.

The next step up in data security implementations is to accelerate software solutions with custom hardware. This is an extremely common approach in x86- and network processor unit (NPU)-based system designs. Figure 1 illustrates a classic coprocessor data security solution.

Coprocessing is an attractive option for its simplicity and ability to elegantly scale the performance of pure software solutions. Its limitation is the performance impact that servicing the coprocessor can impose on the base system's memory – and especially I/O bandwidth. This is less an issue in a dedicated implementation (such as an x86 system packaged as a data security appliance), where you can carefully allocate these resources. And, with the transition from the traditional shared bus architecture of PCI and PCI-X to the switched fabric architecture of PCI Express, coprocessing bandwidth scales considerably and has a bright future.

However, coprocessing is not a panacea, and for many applications an in-line architecture will be more appropriate. This is illustrated in Figure 2.

In-line processors integrate data security directly into the system data path. This is a common approach in communications-centric applications (VPN gateways, storage security appliances) and achieves the equivalent computational off-load as a coprocessor with little to no impact on main system resources.

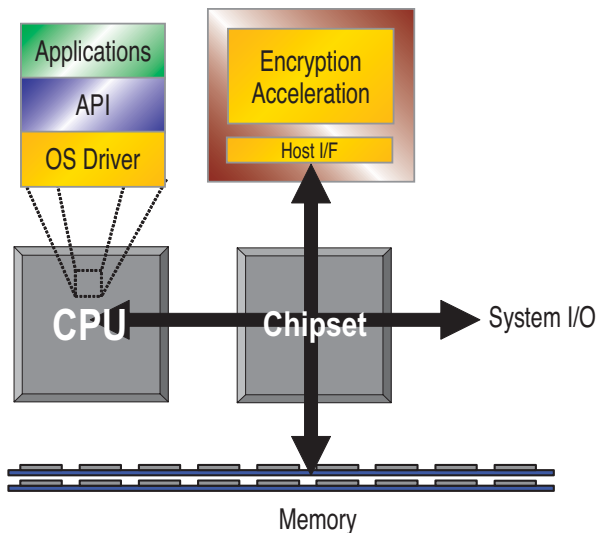
## Data Security Options

To scale performance beyond the abilities of pure software, data security requires additional hardware, either in the form of an add-on coprocessor or in-line data-path processor. So that brings us to the question of what kind of hardware: an ASIC, perhaps an ASSP, or an FPGA.

All of these options have advantages and

disadvantages, and all will be appropriate in certain situations. However, when you consider the number of important decision criteria, the inherent advantages of FPGA solutions can be quite dramatic. To illustrate this, the relative merits of each implementation option across a range of decision criteria are listed in Table 1.

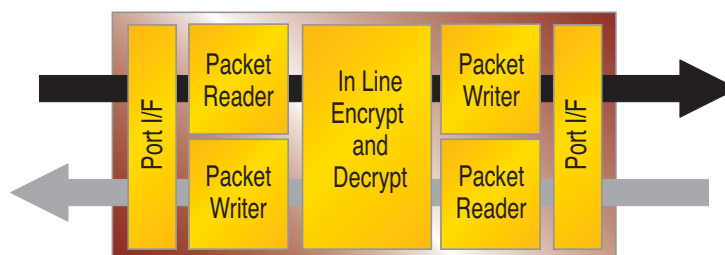
The values in Table 1 present a software-only solution as a baseline against which the three hardware options are compared on a relative basis. Let's explain each of these comparisons.



### Look Aside Co-Processor

Offloads computationally intense workload for another processor

Figure 1 – Look-aside data security coprocessor



### In Line Processor

Performs encryption as a flow-through function in the data path

Figure 2 – In-line data security processing

## NRE

Incremental NRE (non-recurring engineering) from a pure software solution will be required for any hardware solution. In this category, ASSPs come out on top, as they are reasonably well packaged for specific target applications. However, that advantage is negated if you want to use them in a non-standard way.

FPGAs come in second, as by definition they will involve a circuit engineering exercise. However, the NRE for FPGAs is minimal, as they are the most efficient platform

	S/W Only	ASIC	ASSP	FPGA
Non-Recurring Engineering	N/A	High to Very High	Low	Medium
Performance	Low	High	High	High <sup>1</sup>
Unit Cost	N/A	Low	Medium	Medium
Customization	High	High	Low	High
Scalability	Low	Low	Medium	High
Device Availability	N/A	Medium Term	Medium Term	Long Term
Tamper Resistance	Low	Medium	Low to Medium	High
<b>Re-Programmability</b>	<b>High</b>	<b>Low to None</b>	<b>Low to None</b>	<b>High</b>

<sup>1</sup>With the exception of modular exponentiation, which FPGAs do well, but not efficiently for transaction server-class requirements.

Table 1 – Comparison of data security implementation options

for hardware design efforts. ASICs lag far behind, with typically much higher IP licensing costs, mask charges, tool chain costs, and long cycle times.

### Performance

All of the hardware options will be dramatically superior to software only when it comes to performance. Although differences across these options do exist, it is likely not significant. All can deliver essentially wire-speed solutions for almost any application.

The one exception to this rule is modular exponentiation, a function heavily used in public key algorithms. FPGAs are quite capable of supporting 50-1,000 transactions per second (TPS) but do not scale efficiently to far higher performance. For those select applications really requiring 5K+ TPS, you will be better served by an ASIC or ASSP.

### Unit Cost

In terms of physical unit cost, ASICs will obviously win, presuming that you have the necessary unit volumes against which to amortize the NRE. Because of this, ASICs are generally most compelling for high-volume (100K+) class applications. ASSPs and FPGAs, on the other hand, tend to be most suitable for low- to moderate-volume applications.

Some will argue that FPGAs cannot possibly compete against ASSPs in this category, and if you are comparing 100% equivalent configurations, this would be true.

However, ASSPs are by design overloaded with features – so each configuration will have a large enough market to justify its development. This means that in most applications the system uses only a fraction of the features in any given part. In contrast, efficient FPGA design targets the specific, necessary, and sufficient features that you actually need and is only burdened with their implementation. When comparing a “kitchen sink” ASSP versus a “necessary and sufficient” FPGA design, you will find FPGA platforms to be extremely cost-effective.

### Customization

Customization is a metric where ASICs and FPGAs can totally separate from ASSPs. As customer-defined purpose-built solutions, they have the inherent ability to integrate the specific features you require

tailored to the specific design of your architecture, and in so doing impart an exponential benefit. This can take the form of supporting a legacy algorithm not available in an ASSP (there are many), integrating custom features for IT manageability, or tuning the solution to a specific application. In simple terms, it is about differentiation; developing features, performance, and cost efficiencies that distinguish you from your competition.

To illustrate this point, let’s take a basic AES coprocessor as an example. If all you need is a basic coprocessor, there is no particular advantage, as shown in Figure 3A. However, if you are building this solution to serve a tape archive application, then the opportunity exists to integrate additional features and tune the architecture to provide a superior platform.

In our example, this could be the newly minted AES-GCM encryption cipher combined with in-line compression and block mapping to provide a fully integrated solution with significant differentiation for your platform (illustrated in Figure 3B) versus a product designed with an available ASSP.

### Scalability

Scalability relates to servicing a broad price/performance range with a common

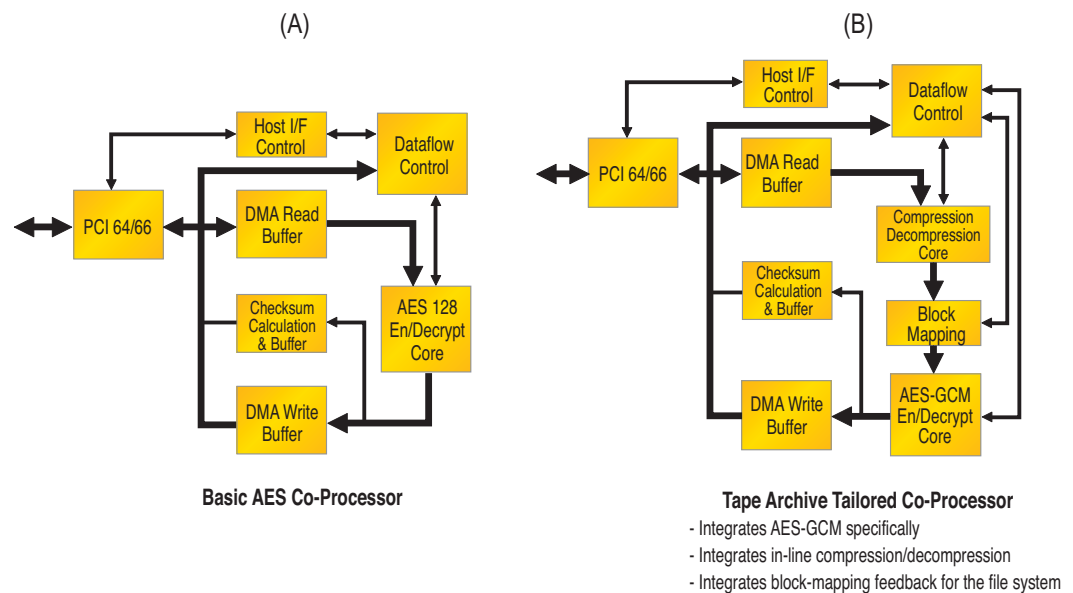


Figure 3 – A basic versus application-customized data security solution

architecture. Software is highly scalable, but as its performance tends to diminish with complexity, it is hard to exploit. One solution is to scale the performance of the host CPU platform, but complexities such as memory type and speed, chipset, and simple cost can make this less than ideal. ASICs do not scale very well either, as they are typically quite targeted by design. ASSPs offer a range of package-compatible price/performance options and thus have reasonable board-level scalability.

FPGAs provide a tremendous dynamic range in price/performance in three distinct ways:

- Custom logic configurations
- A wide range of device options
- Extremely efficient design reuse

In the simplest case, you can support a range of functional options in a single hardware design by simply loading a variety of different logic configurations into the device. In the medium case, you can achieve the same board-level scalability as ASSPs using a range of package-compatible FPGAs, enabling more performance and features. And in the maximum case, you have the ability to re-package a common core of system functionality across a variety of large-scale configurations and efficiently expand your market reach. An example here would be an in-line data-at-rest architecture reworked to support a wide variety of port requirements such as GE, 10GE, Fibre Channel, and SAS to address a range of storage equipment segments.

#### Device Availability

An often important consideration in system design is the long-term procurement logistics for the components used. This can be particularly important in communications infrastructures and aerospace/defense applications where data security is a common requirement. In this regard, FPGAs are second only to software, as they are historically available over very extended commercial lives. ASIC and ASSP availability will vary by vendor, but is typically far more restrictive.

#### Tamper Resistance

In particularly sensitive applications where the hardware is in the field, protecting the architecture against assault can be an important factor too. Data security by definition is deployed to protect valuable content, and therein lies the motivation for extreme attempts to crack a system's architecture.

ASSPs offer some protection in this regard, but as standard parts they are vulnerable to methodical analysis. Anything that is engineered can be reverse engineered; it is simply a matter of effort. ASICs present an additional barrier because of their more limited availability and undocumented nature, but still share similar vulnerabilities. FPGAs stand unique in that when appropriately packaged and safeguarded, they can fully erase themselves if they detect an apparent incursion, and an un-programmed FPGA reveals zero knowledge of your data security algorithms. Admittedly this is an extreme consideration, but one that could be decisive for your most paranoid applications.

#### Reprogrammability

I have saved the best for last, as FPGA reprogrammability can bring truly unique capabilities to your solutions.

The world of data security is perpetually in flux. As standards change, methods evolve, and system and algorithmic vulnerabilities come to light. Software can change, but this is not typically so for ASICs and ASSPs, and almost certainly not at full speed. But FPGAs (field programmable gate arrays) are by definition reprogrammable too, giving them the unique ability to adapt to the changing world over time as efficiently as software. And not just in new product shipments, but in your installed base as well. This is a feature that can take the definition of a managed platform to an entirely new level.

Take for example the disaster of 802.11's WEP security technology. Launched to much acclaim, WEP proved to be amazingly vulnerable to simple attacks. Worse, once this became apparent, it took well over a year for the WPA solution to be defined, new chips

designed and manufactured, and for secure 802.11i products to get to market. To add insult to injury, in the last year WPA has been superseded by WPA2. The result: products sold that did not deliver the security promised and an installed base of products that were left behind – twice.

With an FPGA enabled platform reasonable patches could have been quickly developed, the new specifications deployed immediately upon ratification, and the installed base could have been brought along in the process.

Other examples abound where reprogrammability will prove extremely valuable. Take the continual evolution of cipher modes and authentication algorithms, as seen in IPsec over the last 10 years and soon to continue as the 802.1AE (MACSec) standard emerges. Other examples include IEEE 1619 storage security standards (LRW-AES and AES-GCM) and their inevitable derivatives, as well as some 20 other data security modes currently at the proposal stage with NIST (National Institute for Standards and Technology). See <http://csrc.nist.gov/cryptoolkit/modes/proposedmodes/> for details.

FPGA-based solutions give you the competitive advantage to support such developments before your rivals and to retrofit them into your installed base. That is data security differentiation, and that is the power that FPGAs can bring to your designs.

#### Conclusion

FPGAs give you flexibility, scalability, cost-effectiveness, and adaptability. All of these elements address the fundamental challenges facing today's data security designer: achieving performance across a range of products most cost-effectively while keeping your platform current with continuously evolving technology and integrating your product into the managed IT infrastructure.

To learn more about Xilinx FPGAs and IP solutions for data security, visit [www.xilinx.com/esp/security/data\\_security/index.htm](http://www.xilinx.com/esp/security/data_security/index.htm). 