IEEE TRANSACTIONS ON
EMERGING TOPICS
IN COMPUTING

Received 30 April 2013; revised 6 October 2013; accepted 22 December 2013.
Date of publication 5 February 2014; date of current version 7 May 2014.

Digital Object Identifier 10.1109/TETC.2014.2304492

Test Versus Security: Past and Present

JEAN DA ROLT[1], AMITABH DAS[3], GIORGIO DI NATALE[2], MARIE-LISE FLOTTES[2], BRUNO ROUZEYRE[2], AND INGRID VERBAUWHEDE[3]

[1]UFRGS, Porto Alegre 90035-903, Brazil
[2]University of Montpellier II, Montpellier Cedex 5 34095, France
[3]University of Leuven, Leuven 3001, Belgium

CORRESPONDING AUTHOR: J. DA ROLT (darolt@lirmm.fr)

ABSTRACT Cryptographic circuits need to be protected against side-channel attacks, which target their physical attributes while the cryptographic algorithm is in execution. There can be various side-channels, such as power, timing, electromagnetic radiation, fault response, and so on. One such important side-channel is the design-for-testability (DfT) infrastructure present for effective and timely testing of VLSI circuits. The attacker can extract secret information stored on the chip by scanning out test responses against some chosen plaintext inputs. The purpose of this paper is to first present a detailed survey on the state-of-the-art in scan-based side-channel attacks on symmetric and public-key cryptographic hardware implementations, both in the absence and presence of advanced DfT structures, such as test compression and X-masking, which may make the attack difficult. Then, the existing scan attack countermeasures are evaluated for determining their security against known scan attacks. In addition, JTAG vulnerability and security countermeasures are also analyzed as part of the external test interface. A comparative area-timing-security analysis of existing countermeasures at various abstraction levels is presented in order to help an embedded security designer make an informed choice for his intended application.

INDEX TERMS Hardware security, scan-based attacks, test interface misuse, scan attack countermeasures, comparative area-timing-security analysis.

I. INTRODUCTION

Structural testing is one important step in the production of integrated circuits. Indeed, the fabrication of CMOS devices is not a totally controlled process and some of the manufactured chips may not work properly. Testing is therefore essential to sort faulty and good circuits and thus ensure the quality of the products. The increasing test cost of new technologies demands the insertion of test-oriented structures early in the integrated circuit (IC) design cycle, which is called Design-for-Testability (DfT). These structures aims at improving the testability (mainly the capacity to detect the presence of faults), diagnostics, test time and reducing the number of required test pins.

The most common DfT technique is the insertion of scan chains, which increases the observability and the controllability of the circuit's internal nodes, thereby increasing the testability. Nevertheless, malicious users can use the scan chains to observe confidential data stored in devices implementing cryptographic primitives. Therefore, scan chains inserted in secure ICs can be considered as a source of information leakage. However, testing cannot be simply avoided in secure products for two main reasons: first possible non-tested errors may compromise the system's security (testing is a must for obtaining security certificates as described in [1]) and as for any other IC, the test ensures the quality of the product.

Besides the security threat that is involved in scan chains, standard test interfaces such as JTAG and IEEE 1500 can also be maliciously exploited. These test interfaces that were initially developed for testing printed circuit boards (JTAG) or System-on-Chip internal modules (IEEE 1500), can be used nowadays for debugging purposes. Easy access to debug ports and module's test structures can be used by hackers to steal the contents of on-chip memories (intellectual property) and to modify the firmware/software so that the device executes a function which was not initially conceived by the designer. In order to protect intellectual property, the security of these ubiquitous test interfaces must be improved.

This paper provides a survey on security threats imposed by test structures. In section II the principles underlying the attacks that use the scan paths to jeopardize security, the so-called scan-based attacks are described. Additionally, the state-of-art of scan-attacks and respective countermea-

sures are presented. section III describes the misuse of JTAG and IEEE 1500 interfaces. The overview of the state-of-art countermeasures for both threats is presented in section IV and the conclusion is drawn in section V.

II. SCAN-BASED ATTACKS

The insertion of scan chains consists of replacing the flip-flops (FFs) of the design by scan flip-flops (SFFs) and connecting these SFFs into a shift-register, called scan chain. The scan chain is bound to a input pin (scan-in) and to an output pin (scan-out). An extra pin called scan-enable should be added to control the scan chain's data shifting. If the scan-enable is set to 0, the SFFs are connected to the circuit to behave as functionally expected (functional mode). When the scan-enable is set to 1, the SFFs are connected to the scan chain, and the bitstream at the scan-in is shifted in while the data stored in the SFFs is shifted out through the scan-out pin.

By controlling the scan-in and scan-enable inputs and observing the scan-out pin, and attacker can observe confidential data or corrupt internal states. Fig. 1 illustrates the duality between test and security. While the test engineer uses the scan chains to shift-in input patterns and shift-out response vectors, the attacker may shift-out confidential data (observability attacks) and shift-in corrupted data (controllability attacks).

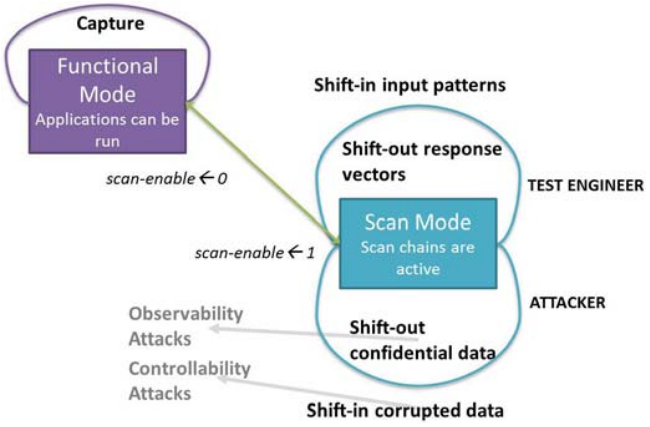

FIGURE 1. Different behaviors: test engineer vs. attacker.

A. ATTACK BASIC PROCEDURE

As depicted in Fig. 1, the attacker can use the shift operation maliciously, switching from functional to test mode at will. Even if the attacker uses the shift operation as the test engineer, the attack's procedure is different from the standard test procedure. For instance, suppose that some of the flip-flops inserted on the scan chain contain confidential information (called hereafter intermediate flip-flops). An observability attack would consist of the following steps: reset the circuit; load the chosen input at the cipher's input; run part of the encryption (functional mode on); switch to test mode when the intermediate flip-flops contain data related to the secret and shift out the scan contents containing this confidential information; analyze the observed contents and try to uncover the secret key. If there is not enough information, repeat the process for another chosen input.

All the known scan-based attacks use this principle to collect scan data. It must be noticed that if the switching from test and functional mode is disabled, then scan-attacks are not feasible. However, this impedes the traditional test procedure. Some countermeasures such as [2] and [3] prevent scan-based attacks by avoiding that unauthorized users can switch between modes.

Some of the known attacks relies on differential analysis. This kind of attack makes use of pairs of cipher inputs and calculates the differences (Hamming distances) from the output related to these pairs. More details on differential scan-based attacks can be found in [4].

This attack procedure can be used to retrieve the value of the secret key of the circuit. If the secret key is stored in flip-flops that are part of the scan chains, it can be directly observed. We can assume that the secret key is stored in non-volatile memories, thus it is not reachable via test structure. However, other registers that store sensitive information related to the secret are certainly inserted in the chain for achieving high testability goals. Therefore, most of the proposed scan-based attacks target registers that store the intermediate results of cryptographic computations, i.e. values dependent on both the input data and the secret key (intermediate registers).

B. ATTACKING CRYPTOGRAPHIC PRIMITIVES

Fig. 2 shows an example of how the scan-based attacks can compromise the security of symmetric-key or public-key cryptography.

Both symmetric-key and public-key algorithms usually have structures that repeat the same operations for multiple iterations. The more iterations, the harder for attackers to find out the secret by only observing the plaintext/input and the ciphertext/output. The algorithmic structure shown in Fig. 2 is simplified, more details can be found in [4]. For symmetric-key algorithms like AES or DES, each intermediate state bit depends on multiple secret bits after one iteration, while each iteration of public-key algorithms like RSA or ECC depends on a single secret bit. Therefore, a scan-attack on symmetric-key implementations needs to focus on the intermediate value obtained after the first iteration. The attacker loads a chosen input and uses the procedure shown in subsection II-A to collect the intermediate data after the first iteration. Collecting a single result resulting from one pair (plaintext, intermediate value) may not be enough to reveal the secret, therefore the attacker collects X results (where X is the number of times the scan-based attack is repeated).

Attacking public-key implementations is slightly different because the attacker has to collect intermediate values in different iterations. For instance, X results (input, intermediate value) are collected for retrieving the first bit of the secret. Then the other secret bits are revealed by shifting out intermediate states after further iterations.

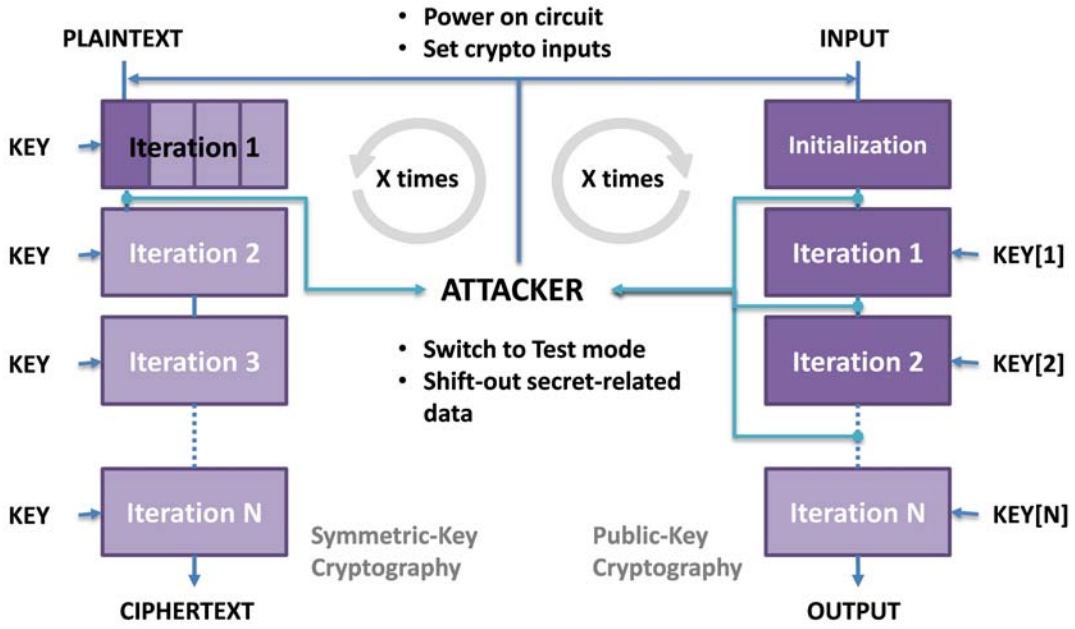

FIGURE 2. Example of scan attack diagrams on SKC and PKC.

The analysis of results (input, intermediate value) that leads to the secret is straightforward. The attacker simulates an execution of the cryptographic core with the corresponding input and the possible values of the secret. In the PKC hardware case, there are only two possible values, *0* or *1*, since one bit of the circuit is attacked at a time. Then the attacker compares the two simulated results and verifies which of them matches the obtained intermediate value, leading to the secret value. If the intermediate value matches both *0* and *1* hypothesis then the attacker is not able to decide which hypothesis is correct, therefore more results should be extracted. The same methodology can be applied to SKC hardware, however the attacker must simulate multiple hypothesis (256 for each AES key byte).

In summary, scan-based attacks are strong because they can observe any intermediate value stored in the scan chains during the execution of cryptographic algorithms in a non-invasive manner.

C. ATTACKER MODEL

In this section, different attacker classes are described. This may help designers in choosing the right countermeasures depending on the target attacker class. Attackers are divided into four classes according to their capabilities:

Class 1: Amateur

- Knows the cipher algorithm implementation, as well as timing diagrams for correctly operating the circuit (this information is usually present in the circuit datasheet);
- Can control the input plaintext, e.g. circuit primary input or external memory location. In other words, this is considered a chosen plaintext attack;
- Can control the scan-enable port and observe the scan-out ports. With this method, the attacker can switch from normal mode to shift mode and vice-versa;

Class 2: Expert

- Can uncover design details with the help of DPA or timing analysis, consisting mainly of input/output register buffers and additional registers that may be affected by plaintext (DFF storage elements). These DFFs may complicate the observation of data related to the secret (further discussed in [4]);

Class 3: Insider

- Knows the correspondence between the circuit flip-flops and their position within the scan chain;
- Thorough knowledge of the DfT structure (i.e. number of scan chains, assignation of flip-flops to the scan chains and order, compaction/decompression);

Class 4: Expert with advanced equipment

- Can remove the chip package and probe internal signals. This is important in cases where the scan chains are disconnected after manufacturing test by means of anti-fuses. This class of attackers can still probe unconnected scan chains;

It must be noted that a Class 3 or 4 attacker have of course all the abilities of the lower class attackers.

D. KNOWN SCAN-BASED ATTACKS

The first scan attack proposed in the literature [5] was conceived to break a Data Encryption Standard (DES) block cipher. Yang et al. described a two-phase procedure that consists in first finding the position of the intermediate registers in the scan chain, and then retrieving the DES first round

key (by applying only 3 chosen plaintexts). In order to find the position, 64 pairs of plaintexts are loaded. Two plaintexts of any of these pairs have a single-bit difference and each pair has a difference in a different location (from bit 1 to 64). Using the procedure described in subsection II-A, the attacker shifts out internal states when the plaintexts are loaded into the registers that store the intermediate values and then these register's flip-flops are localized. Then the attacker applies three chosen plaintexts and shifts out the scan data to recover the first round key. Since the first round key has 48 bits of the secret key (56-bit), the authors propose to perform the same attack on round keys 2 and 3 in order to retrieve the missing bits.

Later the same authors proposed a differential scan attack [6] on the Advanced Encryption Standard (AES). The authors have found that only certain input pairs at the Sbox's input are able to provoke a difference equal to 1 at the Sbox's output. Then the attacker applies some input messages to the AES in hardware, stop the cipher at the first round, shift the scan contents, and calculates the Hamming distances using the differential analysis shown in subsection II-A. The input messages that cause a Hamming distance equal to 1 are then bitwise XORed with the special pairs to obtain a byte of the secret key. By repeating the procedure the attackers can retrieve the whole key. This attack does not require the preliminary step of identifying the position of the intermediate registers.

Some scan attacks have been proposed against stream ciphers. For instance, in [7] the authors suggest that the structure of the Linear Feedback Shift Registers used as stream ciphers can be determined, the seed can be found and then the ciphertext can be unveiled. In order to determine the position of the LFSR bits in the scan chain, the attacker scan out the LFSR state at different clock cycles. Then for each bit, two sets are created: the bits that can be at left and at right of that bit. The attacker supposes that the LFSR polynomial is known and therefore they can simulate which bits can be at left or at right of that bit, using the scan data previously collected. Once all flip-flops are at least in one set, the search continues by choosing one element in each set and repeating the procedure until the position of all bits are unveiled.

Public-key ciphers have also been proven to be susceptible to scan attacks. Binary exponentiation is the target for the RSA scan-attack described in [8], while the Montgomery multiplication method is targeted for the ECC attack [9]. Both attack methods are based on observing the values of the intermediate register of interest using the scan chain. Then correlating this value with a previous offline calculation, which the authors refer to as ''discriminator'', the secret key may be distinguished. Initially all the bits in the scan chain are supposed to store any bit of the intermediate value. Then by shifting out several data obtained from different inputs, the attacker can simulate two possible hypotheses, if the secret key bit is 0 or 1. Then he verifies which bits in the scan chain can match both hypotheses. Repeating this operation multiple times allows the identification of the entire intermediate register.

1) ATTACKS ON ADVANCED DfT STRUCTURES

Scan chains are suitable for increasing testability and thus achieving high fault coverage. However additional techniques are required to solve other issues related to test. For instance, shifting data in and out of larger designs takes too much time, implying higher test costs. Therefore, multiple shorter scan chains are used to meet the required test time. Since the number of test pins is limited, the input test patterns are decompressed to fill several scan chains and the test responses are compressed to meet the few output pins. These structures are called pattern decompressor and response compactor and can be seen in Fig. 3. In addition to these structures, mask decoders are also inserted in the design to filter unknown values (called X-states) that would otherwise corrupt the response compactor output.

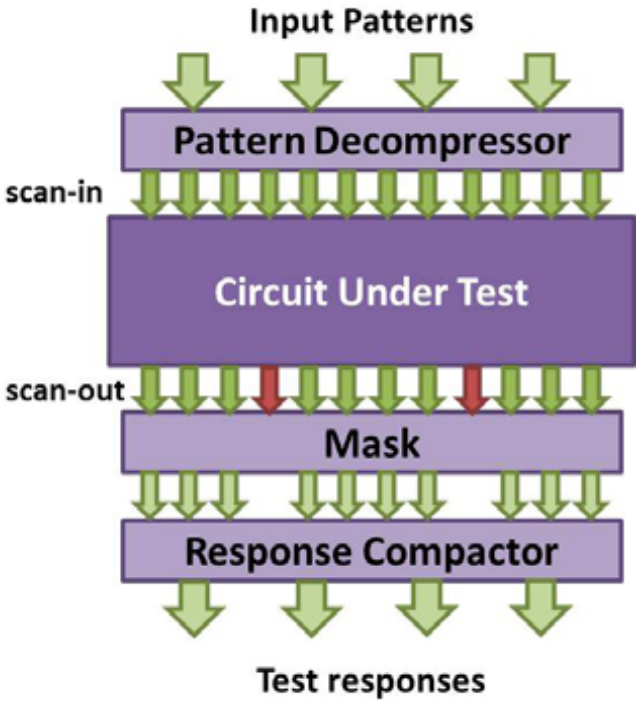

FIGURE 3. Example of design with multiple scan chains, pattern decompressor, response compactor and mask decoder.

The structures shown in Fig. 3 are referred as advanced DfT structures. One example of this suite of advanced DfT structures is the Embedded Deterministic Test (EDT) [10]. EDT's decompressor consists of a ring generator and a phase shifter. The ring generator is a sequential circuit that outputs a vector with a low linear dependency in its outputs while the phase shifter is a combinational circuit that spreads these intermediate outputs to all the internal scan chains. EDT's response compactor can be operated in two different schemes which are selected during test time. The first scheme actually bypasses any decompression and compaction and allows the

tester to load and unload all the scan flip-flop contents individually. This can be achieved by concatenating the internal scan chains into fewer scan chains, directly connected to the inputs and outputs (similar to a design without any advanced DfT structures). The second scheme uses a mask decoder and a linear compactor. First the mask blocks some of the scan outputs from reaching the compactor. This behavior is controlled by a pattern mask, which comes from the input decompressor and can be controlled from the scan inputs. Then, the bits that are not masked pass through a xor-tree that calculates the parity of these bits and outputs the result. If N is the number of test outputs, there are N xor-trees which operate over a different set of internal scan output bits. More details on EDT can be found at [10]. Other industrial suites of advanced DfT structures are similar to EDT and will not be described in this paper. For more details on other techniques, see [11].

In [12] and [13] the authors propose that these advanced DfT structures such as the ones described in subsubsection II-D.1 are resistant to scan-attacks. However, the authors [12] do not consider differential analysis, and therefore they suppose that identifying registers that store secret data is not feasible.

Several attacks have been proposed to show that these advanced DfT structures should not be considered as a countermeasure to scan-based attacks. The first attack that considers these structures [14] extends the attack algorithm proposed in [6] to retrieve the AES secret key in presence of response compaction. However, the authors do not consider complex designs or the mask decoder. In [15], the same authors propose a new attack that targets AES, considering now the presence of mask decoders. This attack exploits properties of the Sbox that allows to recover the secret key by observing test responses that contain only parts of the intermediate register's bits. Moreover, time compaction schemes using Multiple Input Signature Registers (MISRs) are also targeted with the new differential scan attack.

Public-key implementations in circuits that contain advanced DfT structures are also shown to be vulnerable to scan attacks. In [17] the authors describe a procedure to attack RSA implementations by first locating the intermediate register's bits and then retrieving the secret key using distinguishers. The same algorithm is applied to attack an ECC in [18]. In both of them, the attack is applied to netlists containing all the mentioned DfT structures.

A more advanced attack is presented in [4]. This attack, which is based on distuinguishers, is first presented in a generic form and then it is applied to AES, DES, Khazad, ECC and RSA. The authors present other issues present in complex designs that were not considered in the previous attacks, and then show how to deal with these issues, making the attack more realistic.

2) SUMMARY OF SCAN-BASED ATTACKS

The known scan-based attacks that have been proposed are summarized in Table 1. As it can be seen, attacks on stream ciphers in presence of advanced DfT structures has not yet been shown.

TABLE 1. Summary of previous scan-based attacks.

Attack	Target Cipher	Advanced DfT structures		
		SPAT[a]	MASK[a]	TIM[a]
Yang et al.[6]	AES	-	-	-
Yang et al.[5]	DES	-	-	-
Liu et al.[7]	Stream ciphers	-	-	-
Nara et al.[8]	ECC	-	-	-
Nara et al.[9]	RSA	-	-	-
Da Rolt et al.[14]	AES	Yes	-	-
Da Rolt et al.[15]	AES	Yes	Yes	Yes
Da Rolt et al.[17]	ECC	Yes	Yes	-
Da Rolt et al.[18]	RSA	Yes	Yes	-
Da Rolt et al.[4]	AES, DES, Khazad, ECC, RSA, El Gamal	Yes	Yes	Yes

a. SPAT: XOR-tree based spatial compaction. MASK: Mask decoder. TIM: MISR-based timing compaction.

III. MISUSE OF TEST INTERFACES

Test interfaces such as JTAG and IEEE 1500 have two security drawbacks: they make scan-based attacks easier and they can be used to upload corrupted firmware in non-volatile memories or read out internal contents. The first issue comes from the fact that they provide access to individual components (chips on board or cores on SoCs). It implies that malicious users can apply scan-based attacks on the cryptographic blocks only, which makes the analysis phase of the attack easier.

The second issue is illustrated by a well-known example: the first hack of xbox 360 gaming consoles. This hack allows users to run code that was not initially allowed by Microsoft. This can be done by using JTAG to upload a version of the firmware that was hacked. Therefore even if new versions of the firmware are bug-free, the users can always downgrade the firmware to the known insecure version.

Another security flaw of JTAG is related to FPGAs. The configuration bitstream which contains the Intellectual Property (IP) information of a reconfigurable design is mostly programmed via the JTAG interface into FPGAs. The firmware update of set-top boxes used in pay-TV subscriptions also happens in most cases through the JTAG port. An insecure JTAG access would allow on one side to re-program parts of the system at the hacker's will, and on the other side, it could be used to sniff configuration bits thus allowing retrieving the IP information.

Additionally, there have been many practical attacks on secure devices such as set-top box (STB) decoders using the JTAG interface [19]. ARM11 (Cortex) microcontroller, which is used in latest smartphones, has extensive test and debug facilities through the JTAG port. This is a well known backdoor that is currently used for instance to jailbreak iPhones/iPad, or to unlock protected services in mobile phones [20]. Even if not documented, it is reasonable to think

that JTAG could be used to compromise the security of other applications such as mobile e-payments, or Wireless Sensor Nodes (WSNs) [21].

The previous examples of JTAG hacking consider an untrusted user attacking a legitimate device. Nonetheless, fake devices can also steal from authorized users (or service providers). For instance, a fake device may download paid firmware updates from the service provider and use them for free.

IV. OVERVIEW OF STATE-OF-ART COUNTERMEASURES

Several countermeasures have been proposed in the literature and in the industry. We classify them in 3 groups: structures that are offered by DfT tools and may be considered as inherent countermeasures (subsection IV-A); protocol countermeasures which change the test procedure in order to secure the test access (subsection IV-B); and countermeasures that resist against micro-probing of the scan signals (subsection IV-C). subsection IV-B is divided in two parts. First we describe the countermeasures that are extensions of standard test access ports such as JTAG (protecting the circuit against the issues described in section III). Then we present other countermeasures that secure the circuit at protocol-level.

As shown in Fig. 4, the countermeasures described in subsection IV-A and subsection IV-B target attacker classes 1, 2 and 3; while the countermeasures described in subsection IV-C focus on attacker class 4. In order to protect against all attacker classes, at least one countermeasure in each row should be implemented.

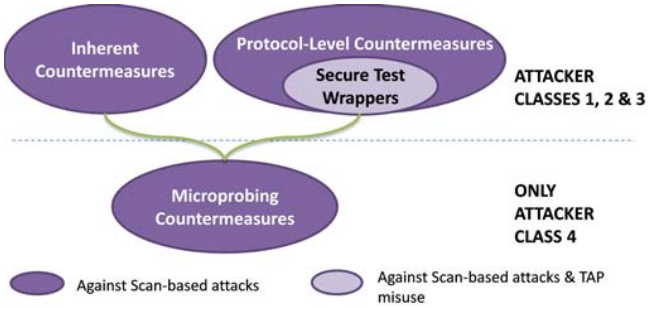

FIGURE 4. Countermeasures categories.

A. INHERENT COUNTERMEASURES

1) ADVANCED DfT STRUCTURES

Besides the attacks listed in subsubsection II-D.1, advanced DfT techniques may not be protect any circuit from scan attacks, particularly small ones where a single scan chain without any compression/compaction circuitry is implemented. Additionally, a common practice is to add a second test mode in large designs, which connects all the internal scan chains and bypass the compression schemes, allowing better fault coverage and diagnosis. An attacker could take advantage of this extra test mode for carrying out ''regular'' scan attacks.

2) BUILT-IN SELF-TEST (BIST)

In order to quickly perform test on standard DfT structures, such as scan chains and advanced DfT structures, Automatic Test Equipment (ATE) is required. However, the increasingly cost of using these test machines is becoming prohibitive. Therefore, including built-in structures to perform at-speed test is a welcome solution, specially because it allows easy online (not concurrent) testing in the field. The most common BIST scheme consists of a pseudo-random generator that feeds the internal scan chains with test patterns, the scan chains, and a output response analyzer that compacts the outputs of the scan chains using space (XOR-tree) and time compaction (Multiple-Input Shift-Register), creating a test signature. If the test signature matches with the signature from a good-simulation (fault-free), then there is no fault in the circuit, otherwise the circuit is faulty. More details on traditional BIST schemes can be found in [22].

From the security point-of-view, BIST ensures that no secret leaks out of the circuit. However, BIST has some issues like low diagnostic resolution and resistance to pseudo-random patterns (impacts on fault coverage), which may be required for some applications.

Alternatively to the standard BIST solution, other ad-hoc solutions were proposed to self-test cryptographic primitives. One of the first examples of cryptographic self-BIST approaches appears in [23] and [24]. In these approaches, the signature generation ability of DES is used for pseudo-random number generation, thereby reducing the hardware cost. In [25] the authors show that by connecting the AES output to its input and running several clock cycles, all the faults are excited, implying a fault coverage of 100 %. Similar results were also proven by Yang et al. [26]. It must be noticed that these solutions work only with block ciphers whose Sbox respect the strict avalanche criterion. Using these ad-hoc solutions on other crypto primitives, like public key ones, is still an open issue.

B. PROTOCOL COUNTERMEASURES

1) SECURE TEST WRAPPERS

These wrappers are inserted around the DUT test interface, and aim at controlling the access to the test infrastructure. Since the JTAG standard is commonly used as interface for test and debug purposes, the secure wrappers tend to be actually a secure version of the JTAG standard.

One of the first approaches for implementing a secure JTAG appears in [27]. It relies on a locking/unlocking mechanism for controlling the access to the JTAG instructions. It is based on storing a secret key inside the chip boundaries. To gain access to the JTAG features the user must shift in the secret key, otherwise the JTAG bypasses all the data on the TDI input to the TDO output. This approach does not consider the case where a fake circuit requests updates that may compromise the intellectual property. Additionally, the key management is an open-issue. If all the circuits share one access key, once a single key is compromised all the devices

are compromised too. If each circuit has one access key, then the service provider/tester must have a database with the correspondence between circuit's ID and access keys. Another problem is the remote access: for upgrading the firmware of a set-top box over the internet, the service provider sends the access key which can be eavesdropped by a malicious user.

A similar approach is presented in [28] where the access is granted after the secret key is shifted-in. This solution targets the IEEE 1500 test wrapper instead of the JTAG. In order to reduce the area overhead, authors propose to reuse the boundary registers as a LFSR that is fed with a secret seed and then generates a larger secret key. This solution faces the same security and key managements issues as in [27].

Another password-based solution was proposed in [29]. The authors suppose a System-on-Chip (SoC) scenario where malicious IPs may steal data from others IPs connected in the same test daisy chain. Therefore this solution targets the IEEE 1500 standard for SoC. The solution requires the addition of a random-key generator to the test controller (inside the circuit). Also, each wrapper must contain a key register to store the keys generated by the test controller. In order to securely distribute the keys, a second daisy chain is created. This second chain consists of a shift register of modified scan cells. Each scan cell is connected to an IP block, and this connection can be enabled or disabled at will. Therefore, to store the generated key into a IP block, the key is shifted through the scan chain and only the scan cell correspondent to that module is enabled. After the initial key setup, the test/debug procedure can proceed using the main daisy chain. If confidentiality of the IP's data is required, then the authors suggest using a stream cipher (the secret key is the one established at the initial key setup) between the wrapper input and output ports.

A detailed evaluation of the JTAG test standard, its security problems, attackers'capabilities, possible attacks and countermeasures has been done in [30]. The authors assume that other ICs on the same board may not be trusted and that they can sniff secret data and corrupt the communication (man-in-the-middle attack). In order to ensure the chip's authenticity, they propose three enhanced secure protocols:

- **Level 1:** Authenticity of the test engineer is ensured by using challenge-response-based protocol. Each chip must be programmed with a unique value, using fuses. After a challenge is shifted in, the circuit calculates a hash (using Trivium stream cipher [31]) using the challenge and the unique value as input. The computation result is sent to the test server as response to the challenge.
- **Level 2:** Ensures the secrecy of the communication between parties by encrypting the stream. Trivium cipher is used in this protocol. Level 2 inherits authenticity from Level 1 protocol.
- **Level 3:** Integrity of the messages is ensured by using a MAC. Level 3 inherits authenticity from Level 1 protocol and secrecy from Level 2.

An anti-tamper JTAG TAP is described in [32] that uses SHA-256 secure hash and a TRNG to create a low-gate overhead challenge/response based access system employing an on-chip internal JTAG P1687 instrument. The authors propose that a set of instructions should be public (do not require to pass the challenge) while others are private. The public JTAG instructions pose no harm to the embedded information while the private instructions may jeopardize the security. Each private instruction must have an independent secret key that is used to generate the good response for the challenge/response authentication.

In [33], the authors propose an IEEE 1500 secure test wrapper that uses a PUF-based challenge-response authentication. The PUF allows the reduction in the area overhead compared to the previous approach [32]. This approach requires an enrollment phase where a trusted server collects the challenge-responses from each circuit and stores it in a database for future use. After the enrollment phase is finished, anti-fuses are blown and the enrolling circuit is disconnected, avoiding modeling attacks.

An elaborate three-party secure JTAG protocol using credentials involving SHA-1 hash algorithm, AES block cipher and several arithmetic operators is presented in [34]. The authors describe the possible attack cases, but the protocol is not proven to be secure.

Secure wrappers based on public-key cryptography have been proposed by [35]. The authors propose to embed an ECC module inside each chip to support a public-key infrastructure. In order to access the circuit, the user must log into a secure server. The circuit sends a challenge to the server which replies with a response. If the response matches the expected value, then the user has a communication session with the device. This scheme verifies that the server and the user are trusted, but it allows fake circuits to obtain illegal updates. The secure protocol proposed by the authors is not referenced and no security proof is presented.

There are also industrial solutions for providing security to the JTAG interface. For instance, ARM Trustzone [36] have an option to disable the JTAG by using anti-fuses. Once disabled the JTAG access is irreversibly blocked. The Secure JTAG Controller (SJC) which features in Freescale Semiconductors i.MX31 and i.MX31L Multimedia Applications Processors is another example. They use an authentication mechanism based on challenge-response. However the CRPs are hardcoded on the device (for area improvement). Storing the CRPs inside the device cannot be considered strong in terms of security due to the storage limitations: observing several challenge-response exchanges can give the attacker enough information to find out all challenge-response pairs.

In [37], the authors implement a secure JTAG mechanism using an enhanced version of ECC-based Schnorr Protocol [38] as the public-key cryptographic protocol to solve the inherent key management problem present in most symmetric-key based approaches. The ECC based Schnorr controller and ECC point multiplier has been integrated with the JTAG interface along with the other modules. This has

been done in a seamless manner so as not to affect the timing aspects of the IEEE 1149.1 JTAG standard, and also keeping the behavior of the TAP finite state machine unchanged. The provably secure Schnorr protocol as well as the ECDSA signature authentication are performed by the Schnorr controller. It interacts with a modified JTAG instruction decoder, ECC module, and a 192-bit Linear Feedback Shift Register acting as a random number generator. The base point coordinates (curve parameters) are fetched from an external non-volatile memory. The system is supposed to be locked in the beginning. In order to unlock it, the tester must manipulate the JTAG inputs to enter the new 'UNLOCK' instruction. Then, the instruction decoder informs the Schnorr controller to start the protocol, by means of an unlock request signal. As soon as the authenticity of the test server is verified, the Schnorr controller releases the lock thereby informing the instruction decoder that other instructions can now be performed. When the system is unlocked, the design under test (DUT) boundary scan register can be controlled. Shifting is always controlled by the test server, and that the timing for executing point multiplications depends on the scalar multiplier. Hardware implementations of the ECC-based Schnorr security controller using Affine and Projective coordinates are also presented along with detailed area and timing results.

2) OTHER COUNTERMEASURES

a) Unbounding

A common technique adopted by smart-card providers is to disable the test circuitry after manufacturing test by blowing anti-fuses located at the ends of the scan chains (such as Actel antifuses [39]). This allows the use of full scan and high quality diagnosis at manufacturing. However, in field maintenance and debug are compromised after incoming test. Additionally, class 4 attackers can use micro-probing to access disconnected scan chains [48].

b) Scrambling & access restriction

In this category, we include countermeasures that require an unlocking phase in order to proceed with the standard test procedure (similar to password-based secure wrappers). In the case the key inserted by the user does not match, these countermeasures can act in two different ways: either they scramble data at the scan output in an unpredictable manner [2], [3], [40], or they simply restrict the observability to the scan pins [41], [42].

Scrambling countermeasures ensure the confusion of the stream shifted out from the scan outputs for unauthorized testers [2], [3], [40]. Authorized testers can either deactivate the scrambling by passing through a password-based authentication or know the scrambling order and can perform the test by correctly reordering the scan output bitstream. However, as presented in [15], scrambling techniques may still leak data dependent on the secret (e.g. through leakage of parity information).

A password-based authentication method is also proposed by Bhunia et al. [41]. In order to proceed with the standard test procedure, the user has to pass through a initial authentication phase with M steps. In each step, the user has to insert the test patterns which contain a subset of N-bits (guess key) that are compared with a golden key. If the user correctly guesses the M different keys, the scan output is observable, otherwise it is blocked. In the case of Lee et al. [42], the scan chain contain k dummy flip-flops. When the user enters a test pattern, the values in the k flip-flops are compared with the respective bits in the pattern. If all the bits match, then the response vector is free to be unloaded, otherwise it is hidden.

These solutions have the same security issues as the password-based secure wrappers [27] and [28].

Another solution based on access restriction is presented in [43]. The authors suppose that the controllability of the scan chain does not present a security leak, and that the observability is the main concern of scan-based attacks. Based on this hypothesis, the authors propose that the test procedure should be slightly changed. Instead of shifting in test patterns, shifting out test responses and comparing the actual responses with the fault-free responses (from a good simulation), in the new procedure, the actual responses are shifted inside the circuit and compared with the actual responses. The comparison is on-chip and thus the scan-out signal is not observable anymore. After each test vector is compared, the result of the comparison can be read at the output (a circuit pin). This scheme has no impact on test time or fault coverage (the same vectors used from ATPG can be reused), but it may reduce diagnostic resolution. The authors also present a solution to add new test vectors in order to improve diagnostic ability, if it is required.

c) Secret-free test

This category contains solutions that allow the use of the test infrastructure, but with data that is not related to the secret. Yang et al. [6] describe a method for securing the test which is based on two modes: secure and insecure. After powering on, the circuit goes to insecure mode where a special key is loaded instead of the actual secret key. This special key is controllable through the scan chain and is used for test purposes only. Once the test procedure is over, the user has to switch the circuit to secure mode. In the secure mode the actual secret key is loaded and the test signals are disabled, avoiding the observability and controlability of contents related to the secret key. This scheme seems to work properly, although the test of the key stored in the mirror-register is not possible.

Da Rolt et al. [44] propose a test solution that forbids scanning out the scan contents when the secret information is still stored at the scan chains. For that purpose the authors use a sensor that counts the number of cycles in functional mode. If the circuit is being used in functional mode for several cycles it means that the scan test is not currently happening, and thus the scan chains may contain data related to the secret. In this case, a shift operation allows only scanning in patterns, but not scanning out responses (the scan-out pin is grounded and the scan data is flushed). On the other hand, if the counter senses that there are few clocks (1 or 2) in functional mode, then the scan procedure may be currently in execution, and the shift operation allows both scanning in and scanning out.

TABLE 2. Countermeasures summary table.

Countermeasure	Testability				Applicability			Security[a]
	Manufacturing facilities		In Mission		Cost	Impact on DfT Flow	Test Procedure	
	Fault Coverage	Diagnosis	Fault Coverage	Diagnosis				
ADVANCED DFT STRUCTURES								
EDT[10]	High	High	High	High	Low	None	Standard	Insecure [b]
BIST								
Tradicional BIST[22]	Depends [c]	Low	Depends[c]	Low	Medium	None	Standard	Resists
Yang et al.[26]	High[d]	Low	High[d]	Low	Low	Low	Similar to standard	Resists
Doulcier et al.[25]	High[d]	Low	High[d]	Low	Low	Low	Similar to standard	Resists
SECURE WRAPPERS[e,f]								
[27], [28], [30], [32], [33]–[35], [37]	High	High	High	High	Depends	Medium	Requires unlocking	Resists
UNBOUNDING								
[39]	High	High	None	None	Antifuses	None	Standard	Resists
SCRAMBLING								
Hely et al. [2]	High	High	High	High	Low	Low	Requires unlocking	Resists[g]
Lee et al. [3], [40]	High	High	High	High	Low	Low	Requires unlocking	Resists[g]
ACCESS RESTRICTION								
Bhunia et al. [41]	High	High	High	High	Low	Low	Requires unlocking	Resists
Lee et al. [42]	High	High	High	High	Low	Low	Requires unlocking	Resists
Da Rolt et al. [43]	High	High	High	Medium/ High	Low	Low	Modified	Resists
SECRET-FREE TEST								
Yang et al. [6]	High	High	High[h]	High	Low	Low	Standard	Resists
Da Rolt et al. [44]	High	High	High	High	Low	Low	Standard[i]	Resists
MODIFIED SCAN CHAIN								
Sengar et al. [45]	High	High	High	High	Low	Low	Standard	Insecure[j]
Fujiwara et al. [47]	High	High	High	High	Low	Low	Standard	Insecure[j]
COUNTERMEASURES AGAINST MICROPROBING								
Hely et al. [2]	High	High	High	High	Low	Medium	Standard	Resists [k]

a. All countermeasures listed in this table are susceptible to microprobing, except for []
b. Vulnerable to new scan attacks such as [4]. Besides, industrial DfT structures often provide a second test mode where we can bypass the decompression/compaction circuitry.
c. Some faults are not reacheable using pseudo-random patterns.
d. High fault coverage was achieved only with block ciphers (mainly AES).
e. See Table 3 and Table 4 for more details on Secure Wrappers.
f. It must be noted that Secure Wrappers have a larger scope than scan-based attacks.
g. It must be noted that this category uses a password-based authentication and thus have the same disavantages as secure wrappers based on the same authentication (See subsubsection 2.4.1).
h. The path between user key and the multiplexer that selects the test key is not tested.
i. The contents for the first shift operation are not scanned out.
j. Vulnerable to differential scan attacks, such as [46].
k. Targets only microprobing attacks which focus on activating the scan-enable signal illegally.

d) Modified scan chain

This category of countermeasures aims at modifying the scan chain by adding unknown logic between the chain links. These solutions consider that the attacker does not know the chain structure (while the tester knows it), which is considered "security by obscurity". Besides they are ineffective against differential attacks, as shown in [46]. In [45] the authors propose the addition of inverters between scan flip-flops, however inverting values is completely useless against differential scan attacks (DSA) such as [6], since the difference of two inverted values is the difference of the value itself. In [47] the authors propose to add XORs between the chain flip-flops. This is equivalent to using local response compaction and thus it does not protect against attackers employing DSA techniques.

C. COUNTERMEASURES AGAINST MICROPROBING

In the intrusion detection subcategory, it is considered that the attacker has bypassed other countermeasures and that he is capable of directly probing the scan chain. Hely et al. [49] propose the use of scan enable trees which detect unauthorized access to the scan enable signal [49]. This technique consists of connecting all the scan flip-flops and the scan enable port to a comparator. If the authentication has been bypassed the scan enable is supposed to be disabled, therefore any illegal shift will raise an alarm by detecting if at least one scan enable is active.

TABLE 3. Secure wrappers summary table.

Wrapper		Timing overhead			Area overhead[a]		Security		
		Initial		Per vector	Hardware Requirements	JTAG Overhead	AUT[b]	CNFD[b]	INT[b]
		JTAG_clk[c]	sys_clk						
PASSWORD-BASED									
Novak et al.[27]		32	0	0	key register	144%	user	no	no
Chiu et al.[28]		256	0	0	key seed	20%	user	no	no
Rosenfeld et al.[29]		64 (per IP)	0	0	Random-key generator, stream cipher (per IP)	38%	user	yes	no
CHALLENGE-RESPOSE BASED on KEYED HASHES									
Rosenfeld et al.[30]	L1[d]	160	1152	0	Trivium, fuses	200%	device	no	no
	L2[d]	160	1152	0	Trivium, fuses	200%	device	yes	no
	L3[d]	160	2304	MAC	Trivium, fuses	471%	device	yes	yes
Clark et al.[32]		256	SHA-256 operation	equal to initial delay	TRNG, SHA-256	>500%	user	no	no
CHALLENGE-RESPOSE BASED on PHYSICAL UNCLONABLE FUNCTIONS									
Das et al.[33]		128	0	0	PUF	60%	user	no	no
CHALLENGE-RESPOSE BASED on SYMMETRIC-KEY CRYPTOGRAPHY									
Park et al.[34]		n.s.[e]	n.s.	n.s.	SHA-1, AES	2246%	user	no	no
CHALLENGE-RESPOSE BASED on PUBLIC-KEY CRYPTOGRAPHY									
Buskey et al.[35]		n.s.	n.s.	n.s.	TRNG, ECC	>2000%	user	no	no
CHALLENGE-RESPOSE BASED on ZERO-KNOWLEDGE PROTOCOLS									
Das et al.[37]	L1[f]	781	240762	0	ECC	1648%	device	no	no
	L2	781	482130	0	ECC	1648%	user	no	no
	L3	973	722892	0	ECC	1648%	both	no	no
	L4	1549	1205216	0	ECC	1648%	both	no	no

a. The area overhead is a rough estimation since solutions include FPGA and ASIC implementations, and some of the required blocks are not included in the original area overhead calculations. We used a 180 gate-equivalent JTAG as base to the comparisons.

b. AUT: Ensures authentication. CNFD: Ensures confidentiality. INT: Ensures integrity.

c. JTAG_clk: number of clock cycles used to shift in and out JTAG vectors. sys_clk: number of system clock cycles.

d. The authors proposed three levels of security: L1, L2 and L3.

e. n.s.: not specified.

f. The authors proposed four levels of security: L1 (device is authenticated), L2 (user is authenticated), L3 (both device and user are authenticated), L4 (Elliptic Curve Digital Signature Algorithm is used to verify the exchanged keys, in the case a trusted authentication server is available)

The same authors proposed another solution for this issue: to add ''spy'' flip-flops in the scan-chain, which detect the unauthorized shifting at mission time. These spy flip-flops are inserted between actual scan flip-flops, and in normal mode they are always loaded with a constant value (for instance 0). Then the outputs of these flip-flops are connected to a comparator that senses if they store the same constant value. Illegal shifts will eventually (probably) load these flip-flops with a different value, allowing for intrusion detection.

It must be noticed that these solutions cannot stand on their own: they always require another countermeasures such as a secure wrapper in order to block the test mode to unauthorized users.

D. SUMMARY OF COUNTERMEASURES

The countermeasures against scan-based attacks are summarized in Table 2. It summarizes various aspects of the countermeasures. It first considers the fault coverage and diagnosis in both manufacturing facilities and in mission (incoming test or in-field test). The applicability is divided in cost (area overhead or additional infrastructure), DfT flow (easy to insert) and test procedure. Table 2 also shows comments on the security of each solution.

In order to provide a guideline to designers, the next subsections list the solutions to consider depending on some design requirements.

1) IS SECURITY A REQUIREMENT?

Some of the countermeasures in Table 2 does not respect this requirement. For example, industrial DfT structures have not been conceived for security purposes, and thus cannot withstand new scan-based attacks. Besides, bypassing advanced DfT structures is offered as a secondary test mode for diagnostic purposes, which allows scan-attacks on a single scan-chain that contains all scan flip-flops.

The Modified Scan Chain category (see Table 2) is also considered insecure, because it is based on security-by-obscurity. Apart from this, they may be susceptible to differential scan-attacks, as shown in [46].

2) DIAGNOSTIC RESOLUTION IS NOT A REQUIREMENT

If diagnostic resolution is not a requirement and the circuit can achieve high fault coverage with pseudo-random patterns, then BIST is welcome to counter scan attacks. Besides, BIST entails other advantages such as easy in-field test and at-speed test. Additionally, if a block cipher is present in the design,

it can be used to replace part of the BIST circuitry (area reuse) as pseudo-random generator and response compaction, as suggested in [26]. In case the block cipher is the only block with confidential data, it can be self-tested with an ad-hoc solution as proposed in [25].

3) IN-FIELD TEST IS NOT A REQUIREMENT

In this case, the most simple solution is to use anti-fuses (unbounding), if they are available.

4) ARE IN-FIELD TEST AND DIAGNOSTIC RESOLUTION REQUIREMENTS?

In this case, three categories of solutions meet the requirements: Secure Wrappers, Access Restriction and Secret-Free Test. As we mentioned, secure wrappers are normally used in the presence of a test access port, such as JTAG or IEEE 1500. Besides, they protect against other issues, such as described in subsubsection II-D.1. If these other threats are not expected, then the designer should use either a Access Restriction, Scrambling or a Secret-Free Test solution.

[3], [40]–[42] requires the use of a shared secret-key between user and device, and thus it implies an initial unlocking step. [43] requires no shared secret-key, but it needs extra vectors for improving the diagnostic resolution. Other solutions that should be considered in this case are [6] and [44].

5) ARE MICROPROBING ATTACKS A CONCERN?

In this case, techniques such as [2] should be used together with other regular countermeasures. However, it must be noted that high-end crypto circuits, such as smart cards, already contain countermeasures to detect invasive attacks such as microprobing. In this case no additional technique is required.

E. SUMMARY OF SECURE TEST WRAPPERS

Since the goal of secure test wrappers (STWs) is to cope with other threats besides scan-based attacks (i.e. IP stealing, cloning), we describe secure wrapper solutions separately in this section. In order to summarize the contributions of all solutions, we prepared two summary tables. Table 3 contains timing and area overhead, as well as security aspects and Table 4 displays the infrastructure cost summary.

The first thing to be noted from Table 3 is that the area overhead is high (most of the time greater the 100%). This comes from the fact that the JTAG or IEEE 1500 wrappers are very small (<200 slices). For this reason, most of the papers that present secure wrappers calculate the area overhead considering larger designs such as an AES or a microprocessor. Therefore for larger designs, even solutions [34] and [35] may have a small area overhead. Concerning the hardware requirements, some solutions rely on PUFs and TRNGs where IPs may be bought from third-party designers.

With regard to security, password-based authentication should be used only when the environment is controlled, otherwise replay attacks or a man-in-the-middle can compromise the security. The challenge-response based authentication is enough in most cases. When security must attain high levels, the zero-knowledge solutions are advised.

The main issue with the proposed solutions is that a secret key must be shared between user/tester and device. For that purpose, either the key is transmitted hand-to-hand with the device, leading to insecure management or there is a secure server that manages the keys.

TABLE 4. Infrastructure Cost.

Wrapper type	Circuit Personalization	Key Management	3rd-Party required?
Novak et al.[27]	Store secret key	Shared-key	No [a]
Chiu et al.[28]	Store secret key	Shared-key	No [a]
Rosenfeld et al.[30]	Store secret key	Shared-key	No [a]
Clark et al. [32]	Store secret key	Shared-key	No [a]
Das et al.[33]	Enrollment phase to obtain CRP[b]	Shared CRP	No [a]
Park et al.[34]	Store secret key	Shared-key	Yes [c]
Buskey et al.[35]	Store server public-key	Shared-key	Yes [c]
Das et al.[37]	Store server public-key	Shared-key	Yes [c]

a. To obtain the secret, if the user does not have it. In this case a secure channel between user and server must be created.
b. Challenge-response pairs
c. To authenticate user credentials.

V. CONCLUSION

In this paper we described two main issues related to the test and security domain: scan-based attacks and misuse of JTAG interfaces. Both threats exploit security issues present in structures that implemented test and debug of digital ICs. To help the understanding of scan-based attacks, we have described the principles of these attacks. Then we presented a survey of the known scan-based attacks so that designers can take them into account when building new circuits. Additionally, we described some well known issues related to the misuse of JTAG and IEEE 1500 test interfaces.

In order to help designers to chose the right countermeasures to cope with both threats, we described the solutions presented in the literature. Summary tables have been compiled highlighting the advantages and disadvantages of these countermeasures. Besides the designers, researchers can also make use of this survey to build more sophisticated solutions that consider both security and testability.

REFERENCES

[1] (1994). *Federal Information Processing Standards Publication 140-2: Security Requirements for Cryptographic Modules* [Online]. Available: http://csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf
[2] D. Hely, M.-L. Flottes, F. Bancel, B. Rouzeyre, N. Berard, and M. Renovell, "Scan design and secure chip [secure IC testing]," in *Proc. 10th IEEE IOLTS*, Jul. 2004, pp. 219–224.
[3] J. Lee, M. Tehranipoor, C. Patel, and J. Plusquellic, "Securing designs against scan-based side-channel attacks," *IEEE Trans. Dependable Secure Comput.*, vol. 4, no. 4, pp. 325–336, Oct. 2007.
[4] J. Da Rolt, G. Di Natale, M. Flottes, and B. Rouzeyre, "A novel differential scan attack on advanced DFT structures," *ACM Trans. Des. Autom. Electron. Syst.*, vol. 18, no. 4, p. 58, Oct. 2013.

[5] B. Yang, K. Wu, and R. Karri, "Scan based side channel attack on dedicated hardware implementations of data encryption standard," in *Proc. ITC*, Oct. 2004, pp. 339–344.

[6] B. Yang, K. Wu, and R. Karri, "Secure scan: A design-for-test architecture for crypto chips," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 25, no. 10, pp. 2287–2293, Oct. 2006.

[7] Y. Liu, K. Wu, and R. Karri, "Scan-based attacks on linear feedback shift register based stream ciphers," *ACM Trans. Des. Autom. Electron. Syst.*, vol. 16, no. 2, pp. 1–15, Mar. 2011.

[8] R. Nara, K. Satoh, M. Yanagisawa, T. Ohtsuki, and N. Togawa, "Scan-based side-channel attack against RSA cryptosystems using scan signatures," *IEICE Trans. Fundam. Electron., Commun. Comput. Sci.*, vol. E93-A, no. 12, pp. 2481–2489, Dec. 2010.

[9] R. Nara, N. Togawa, M. Yanagisawa, and T. Ohtsuki, "Scan-based attack against elliptic curve cryptosystems," in *Proc. 15th ASP-DAC*, Jan. 2010, pp. 407–412.

[10] J. Rajski, J. Tyszer, M. Kassab, and N. Mukherjee, "Embedded deterministic test," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 23, no. 5, pp. 776–792, May 2004.

[11] L.-T. Wang, C.-W. Wu, and X. Wen, *VLSI Test Principles and Architectures: Design for Testability* (Systems on Silicon). San Mateo, CA, USA: Morgan Kaufmann, 2006.

[12] C. Liu and Y. Huang, "Effects of embedded decompression and compaction architectures on side-channel attack resistance," in *Proc. 25th IEEE VLSI Test Symp.*, 2007, pp. 461–468.

[13] S. T. Mentor Graphics and Y. A. Whitepaper, *High Quality Test Solutions for Secure Applications*. Wilsonville, OR, USA: Mentor Graph. Corp., Apr. 2010.

[14] J. Da Rolt, G. Di Natale, M. Flottes, and B. Rouzeyre, "Scan attacks and countermeasures in presence of scan response compactors," in *Proc. 16th IEEE ETS*, May 2011, pp. 19–24.

[15] J. D. Rolt, G. D. Natale, M. Flottes, and B. Rouzeyre, "Are advanced DFT structures sufficient for preventing scan-attacks?" in *Proc. IEEE 30th VLSI Test Symp.*, Apr. 2012, pp. 246–251.

[16] B. Ege, A. Das, L. Batina, and I. Verbauwhede, "Security of countermeasures against state-of-the-art differential scan attacks," in *TRUDEVICE*. Nijmegen, The Netherlands: Radboud University Nijmegen, 2013.

[17] J. Da Rolt, A. Das, G. Di Natale, M. Flottes, B. Rouzeyre, and I. Verbauwhede, "A new scan attack on RSA in presence of industrial countermeasures," in *Proc. 3rd Int. Conf. Construct. Side-Channel Anal. Secure Des.*, May 2012, pp. 89–104.

[18] J. Da Rolt, A. Das, G. Di Natale, M. Flottes, B. Rouzeyre, and I. Verbauwhede, "A scan-based attack on elliptic curve cryptosystems in presence of industrial design-for-testability structures," in *Proc. IEEE Int. Symp. DFT VLSI Nanotechnol. Syst.*, Oct. 2012, pp. 43–48.

[19] (2012). *Dishnet: In House Made with Locking Script* [Online]. Available: http://www.satcardsrus.com/dish_net%203m.htm

[20] L. Greenemeier, *iPhone Hacks Annoy AT&T but Are Unlikely to Bruise Apple*. San Francisco, CA, USA: Scientific, Aug. 2007.

[21] K. Eagles, K. Markantonakis, and K. Mayes, "A comparative analysis of common threats, vulnerabilities, attacks and countermeasures within smart card and wireless sensor network node technologies," in *Proc. 1st IFIP TC6/WG8.8/WG11.2 Int. Conf. Inf. Sec. Theory Pract., Smart Cards, Mobile Ubiquitous Comput. Syst.*, May 2007, pp. 161–174.

[22] P. Wohl, J. A. Waicukauski, and S. Patel, "Scalable selector architecture for X-tolerant deterministic BIST," in *Proc. 41st Annu. Des. Autom. Conf.*, Jul. 2004, pp. 934–939.

[23] I. Verbauwhede, F. Hoornaert, J. Vandewalle, and H. De Man, "Security considerations in the design and implementation of a new DES chip," in *Proc. Advances in Cryptology—EUROCRYPT* (Lecture Notes in Computer Science). New York, NY, USA: Springer-Verlag, 1987, pp. 287–300.

[24] I. Verbauwhede, "VLSI design methodologies for application-specific cryptographic and algebraic systems," Ph.D. dissertation, Dept. Elektrotech., Katholieke Univ. Leuven, Faculty Eng., Leuven, Belgium, Jul. 1991.

[25] G. Di Natale, M. Doulcier, M.-L. Flottes, and B. Rouzeyre, "Self-test techniques for crypto-devices," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 18, no. 2, pp. 329–333, Feb. 2010.

[26] B. Yang, "Design and test for high speed cryptographic architecture," Ph.D. dissertation, Dept. Electr. Comput. Eng., New York Univ., New York, NY, USA, 2009.

[27] F. Novak and A. Biasizzo, "Security extension for IEEE Std 1149.1," *J. Electron. Test.*, vol. 22, no. 3, pp. 301–303, Jun. 2006.

[28] G.-M. Chiu and J. C.-M. Li, "A secure test wrapper design against internal and boundary scan attacks for embedded cores," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 20, no. 1, pp. 126–134, Jan. 2012.

[29] K. Rosenfeld and R. Karri, "Security-aware SoC test access mechanisms," in *Proc. IEEE 29th VLSI Test Symp.*, May 2011, pp. 100–104.

[30] K. Rosenfeld and R. Karri, "Attacks and defenses for JTAG," *IEEE Des. Test*, vol. 27, no. 1, pp. 36–47, Jan. 2010.

[31] C. De Cannière, "Trivium: A stream cipher construction inspired by block cipher design principles," in *Proc. 9th Int. Conf. Inf. Sec.*, Sep. 2006, pp. 171–186.

[32] C. Clark, "Anti-tamper JTAG TAP design enables DRM to JTAG registers and P1687 on-chip instruments," in *Proc. IEEE Int. Symp. HOST*, Jun. 2010, pp. 19–24.

[33] A. Das, U. Kocabas, A. S. Sadeghi, and I. Verbauwhede, "PUF-based secure test wrapper design for cryptographic SoC testing," in *Proc. Des., Autom. Test Eur. Conf. Exhibit.*, Mar. 2012, pp. 866–869.

[34] K. Park, S. G. Yoo, T. Kim, and J. Kim, "JTAG security system based on credentials," *J. Electron. Test.*, vol. 26, no. 5, pp. 549–557, Oct. 2010.

[35] R. Buskey and B. Frosik, "Protected JTAG," in *Proc. Int. Conf. Workshops Parallel Process.*, 2006, pp. 405–414.

[36] ARM. Cambridge, U.K. (2012). *ARM Trustzone* [Online]. Available: http://www.atmel.com/Images/doc8558.pdf

[37] A. Das, J. Da Rolt, S. Ghosh, S. Seys, S. Dupuis, G. Di Natale, *et al.*, "Secure JTAG implementation using Schnorr protocol," *J. Electron. Test., Theory Appl.*, vol. 29, no. 2, pp. 193–209, Apr. 2013.

[38] C. Schnorr, "Efficient identification and signatures for smart cards," in *Advances in Cryptology*, G Brassard, Ed. New York, NY, USA: Springer-Verlag, 1989, pp. 239–252.

[39] Actel. Mountain View, CA, USA. (2002). *Design Security in Nonvolatile Flash and Antifuse Fpgas* [Online]. Available: http://www.actel.com/documents/DesignSecurity_WP.pdf

[40] J. Lee, M. Tehranipoor, C. Patel, and J. Plusquellic, "Securing scan design using lock and key technique," in *Proc. 20th IEEE Int. Symp. DFT VLSI Syst.*, Oct. 2005, pp. 51–62.

[41] S. Paul, R. Chakraborty, and S. Bhunia, "VIm-scan: A low overhead scan design approach for protection of secret key in scan-based secure chips," in *Proc. 25th IEEE VLSI Test Symp.*, May 2007, pp. 455–460.

[42] J. Lee, M. Tebranipoor, and J. Plusquellic, "A low-cost solution for protecting IPs against scan-based side-channel attacks," in *Proc. 24th IEEE VLSI Test Symp.*, May 2006, pp. 1–6.

[43] J. D. Rolt, G. D. Natale, M.-L. Flottes, and B. Rouzeyre, "Thwarting scan-based attacks on secure-ICs with on-chip comparison," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. PP, no. 99, p. 1.

[44] J. Da Rolt, G. Di Natale, M.-L. Flottes, and B. Rouzeyre, "A smart test controller for scan chains in secure circuits," in *Proc. IEEE 19th IOLTS*, Jul. 2013, pp. 228–229.

[45] G. Sengar, D. Mukhopadhyay, and D. R. Chowdhury, "Secured flipped scan-chain model for crypto-architecture," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 26, no. 11, pp. 2080–2084, Nov. 2007.

[46] J. Da Rolt, G. Di Natale, M. Flottes, and B. Rouzeyre, "New security threats against chips containing scan chain structures," in *Proc. IEEE Int. Symp. HOST*, Jun. 2011, p. 110.

[47] H. Fujiwara and M. E. J. Obien, "Secure and testable scan design using extended de Bruijn graphs," in *Proc. 15th ASP-DAC*, Jan. 2010, pp. 413–418.

[48] O. Kömmerling and M. G. Kuhn, "Design principles for tamper-resistant smartcard processors," in *Proc. USENIX Workshop Smartcard Technol. USENIX Workshop Smartcard Technol.*, 1999, pp. 1–2.

[49] D. Hely, F. Bancel, M.-L. Flottes, and B. Rouzeyre, "Secure scan techniques: A comparison," in *Proc. 12th IEEE ISOLT*, Jan. 2006, pp. 119–124.

JEAN DA ROLT received the B.S degree in computer engineering from the Institute of Informatics, Universidade Federal do Rio Grande do Sul, Porto Alegre, Brazil, in 2008, the B.S degree in telecommunications engineering from the École Nationale Suprieure d'Informatique et Mathmatiques Appliques, Grenoble, France, in 2008, and the Ph.D. degree with the Laboratoire dInformatique, de Robotique et de Microlectronique de Montpellier, Montpellier, France. His current research interests include cryptography, electronic design and automation and telecommunications.

AMITABH DAS received the B.Tech degree in electronics and communication engineering from Kalyani Government Engineering College, West Bengal, India, in 2003, and the M.Tech degree in instrumentation and electronics engineering from Jadavpur University, Kolkata, India, in 2009. From 2003 to 2007, he was as an Assistant Systems Engineer with Tata Consultancy Services, Kolkata, India, and as an Electronics and Instrumentation Engineer with Bharat Heavy Electricals Limited, India. He received the Ph.D. degree in the Department of Electrical Engineering from the KU Leuven, Belgium. His current research interests include secure design-for-testability, electronic design automation, hardware cryptography, embedded systems, and hardware/software co-design and co-simulation.

GIORGIO DI NATALE received the Ph.D. degree in computer engineering from the Politecnico di Torino, Italy, in 2003. Currently, he is a Researcher with the French National Research Center, Laboratoire dInformatique, de Robotique et de Microlectronique de Montpellier, Montpellier. He has published articles in publications spanning a broad range of diverse disciplines, including memory testing, fault tolerance, software implemented hardware fault tolerance and secure circuits. He serves the Test Technology Technical Council of IEEE Computer Society as Web Master.

MARIE-LISE FLOTTES received the Ph.D. degree in electrical engineering from the University of Montpellier, in 1990. She is currently a Researcher with the French National Research Center. Since 1990, she has been conducting research in the domain of digital system testing with the Laboratoire dInformatique, de Robotique et de Microlectronique de Montpellier, France. His current research interests include design for testability, testability and dependability of secure circuits, test data compression and test management for SoC and SiP.

BRUNO ROUZEYRE received the master's degree in mathematics in 1978, Ph.D. degree in CAD in 1984, from the University of Montpellier. Currently, he is a Professor with the University of Montpellier and conducts his research with the Laboratoire dInformatique, de Robotique et de Microlectronique de Montpellier. His current research interests include several aspects of CAD for digital circuits and are particularly oriented toward optimization, verification, test and test synthesis of digital and secure circuits.

INGRID VERBAUWHEDE received the electrical engineering and Ph.D. degrees from the KU Leuven, Belgium, in 1991. From 1992 to 1994, she was a Post-Doctoral Researcher and Visiting Lecturer with the University of California, Berkeley. From 1994 to 1998, she worked for TCSI and ATMEL in Berkeley, California. In 1998, she joined the Faculty with the University of California, Los Angeles (UCLA). She is currently a Professor with the KU Leuven and an Adjunct Professor at UCLA. At KU Leuven, she is a Co-Director with the Computer Security and Industrial Cryptography Laboratory. Her current research interests include circuits, processor architectures and design methodologies for real-time embedded systems for security, cryptography, digital signal processing, and wireless communications. This includes the influence of new technologies and new circuit solutions on the design of nextgeneration systems on chip. She was the Program Chair of CHES07, ASAP08, and ISLPED02. She was also the General Chair of ISLPED03 and CHES12. She was a Member of the executive committee of DAC05 and 06.