Secure Mechanism for Remote Bitstream Updates of Reconfigurable Computing Platform

Run-feng Huang
School of Information Engineering,
Zhengzhou University
Zhengzhou, China
e-mail: hrunfeng@gmail.com

Qing-lei Zhou
School of Information Engineering,
Zhengzhou University
Zhengzhou, China
e-mail: ieqlzhou@zzu.edu.cn

***Abstract*—Although FPGA-based reconfiguration computing platform has been shown as a promising technique in high productivity computing field, it may suffer security threats during its bitstream remote update process. A secure mechanism is proposed in this paper to prevent tamper and replay attacks during the process of remote bitstream update on reconfigurable computing platform， which encrypts the bitstream and calculates the value of MAC when making the configuration at the Remote Update Server. And then these data is sent to the OMC(Operation-Management-Control) Units on reconfigurable computing platform via network. , The encrypted bitstream will be sent to FPGA after its integrity was verified by the value of MAC. At the end of the secure update process, FPGAs will decrypt the encrypted bitstream and store result in RAM which is in more sercure boundary, subsequently FPGA will be reconfigurated with the no-encrypted bitstream in the RAM. Security analysis shows that the scheme proposed can guarantee the integrity and the confidentiality of the bitstream for remote updating and also has the ability of protecting the updating process from replay attacks, at the same time no extra FPGA logic resources will be used.**

Key words: Security mechanism; FPGA; Reconfigurable technology; Remote bitstream update; High-performance computing;

I. INTRODUCTION

With the development of high performance computing, faults of super computer gradually revealed in terms of power consumption. Generally the power consumption of supercomputer systemsare more than 5000 kw today, high performance bringsalong with high energy consumption, such as Google's cloud computing center, its electricity consumption a day and night is equalto the Geneva, Switzerland, which has sparked the doubt about the input-output ratio of high performance computing system. In recent years, the focus of recent research in the field of high-end Computing aretransferredfrom High performance Computing to gradually to the High Productivity Computing. Adopting reconfiguration computing platform, according to different applications computing resources is reconfigured, which can effectively improve the computation productivity. Reconfiguration calculation refers to under the control of software, using reconfigurable computing resources in the system, according to the characteristics of the application to build a new logical structure of the calculation for the application, reaching closely to the high-performance computing of dedicated hardware design mode[2]. Reconfiguration technology based on Field Programmable Gate arrays (FPGA) is one of the key technologies of reconfiguration calculation system, this paper mainly studies the secure update scheme of remote bitstream under the SRAM FPGA reconfiguration calculation device.

FPGA is a kind of a signal processing device which can be used in programmable way, the user can its function be defined by users through changing the configuration information of it, which could meet the design requirements. Compared with the traditional digital circuit system, FPGA has the advantage of being programmable, high integration, high speed and high reliability etc. And the most mainstream FPGA is based on SRAM technology, but after the SRAM FPGA powered off, FPGA will lose all logical relationship, so it is generally equipped with Non-Volatile Memory (NVM) outside of chip, which is used to save the configuration information during power outages, but outside of chip NVM easily is attacked by opponent, resulting in information leakage of the bitstream.

In regard to remote update about the FPGA bitstream and secure transmission, Adietc[3] and Mr Castilloetc[4] proposes two different bitstream update security plan, but are not perfect and safe. In the literature [5], Drimer describes a novel and effective remote update security protocols for the FPGA reconfiguration, but does not give security analysis, and decryption circuit takes up a lot of resources of the FPGA configuration logic, which is not conducive to the design of user logic and performance improvement. Literature [6] proposed a bitstream update plan which can prevent against replay attack, but does not provide a guarantee scheme on the bitstream confidentiality and integrity.

In this paper a remote bitstream security update mechanism is proposed under the SFRCP platform. Under

Fund program: National 863 Program key program (program number: 2009AA012201).

this mechanism, the configuration information is stored in encrypted way in NVM which is in SFRCP control unit and on the remote update server-side bitstream is encrypted and the MAC value is calculated to ensure the confidentiality and integrity of the bitstream, at the same time the decrypted bitstream is stored in the RAM, and then secondary configuration of FPGA so as to reduce resource utilization on the FPGA which is used by the decrypt logic, and taking advantage of storage in high security boundary to store bitstream can reduce the communication traffic between SFRCP control unit and the update server. In addition, using a variety of encryption logics the safety of the bitstream could increase. In section 3 we give the security analysis for the solutions.

II. THE RECONFIGURABLE COMPUTING PLATFORM

A. *Architecture of The SFRCP*

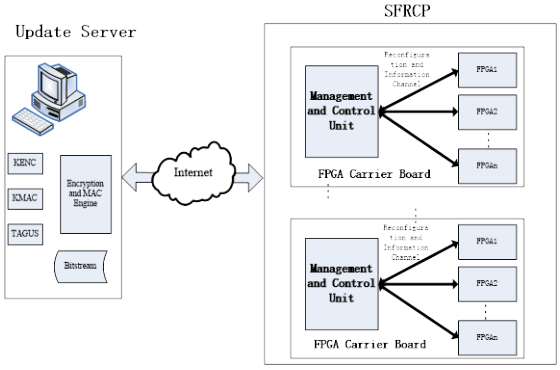

Figure 1. Architeture of the SFRCP

The architecture of a single calculation component of SFRCP system is shown in figure 1. The resource management module of remote update server is responsible for generating the new configuration requirements according to the application requirements of the SFRCP side. New configuration is encapsulated in the bitstream data packets, on the update server bitstream is encrypted with the key in the key storage and subsequently MAC values are calculated, those data is sent to the control unit of SFRCP side through the insecure Ethernet which does not guarantee the security of the information. After control and management unit verify the MAC value correctly, encrypted bitstream should be stored into the NVM. When the FPGA power on at the bottom or need to update, the Management and Control unit according to the FPGA ID information in bitstream packet to determine the corresponding FPGA which the bitstream will be downloaded in, thus the remote update of bitstream accomplishes.

SFRCP side has multiple reconfigurable components and each is composed of multiple FPGAs, each FPGA equipped with SRAM, DDR3 memory. Information collection is run on SFRCP-end and dynamic reconfiguration is mainly rely on the control and management module within reconfigurable components, control and management module is responsible for collecting work status and internal information of the FPGA, and will report the remote update server with the collected information to provide the basis for the resource management and allocation.

B. *Security Threat of The Rconfiguratoin on The SFRCP*

The dynamic reconfiguration methods based on FPGAs basically are as follow: off-chip reconfiguration[1], the reconfiguration based on modular circuit design and reconfiguration based on the technology of the bitstream. Based on bitstream technology of dynamic reconfiguration is the main research direction, the bitstream update mechanism is the important means to realize dynamic update of the configuration for the FPGA reconfiguration.

SFRCP platform using the reconfiguration method based on bitstream technology, its security threats mainly derived from the data and the IP(Intellectual Property) in the SRAM FPGA which the attacker attempt to attack. So as to ensure the safety of the S FRCP platform is mainly to ensure data security and the FPGA IP core of FPGA's safety. In detail, the FPGA's data security includes protecting the data of the FPGA's application circuit and the data security in the process of communication from a peripheral circuits in and out. Protecting the security of the FPGA IP will be expected to protect the FPGA bitstream information, in order to prevent the design being attacked by cloning or reverse [7]. In this article we focuses on the remote update problems of FPGA bitstream, the main security threats are described as follow:

1) Wiretap: the attacker can use the network monitoring tools to steal the FPGA system design.

2) Tamper: the attacker manipulate the data transmitted through the network to deceive the FPGA system.

3) Replay: the attacker replay the early intercepted information to obtain the trust of the FPGA or degrade the configuration of the FPGA system.

Therefore the design of bitstream update mechanism needs to be integrated by encryption and Hash security measures to ensure information confidentiality and integrity and so on, also need to ensure the effective use of FPGA resources.

III. PREPARE THE SECURITY MECHANISM OF THE FPGA REMOTE RCONFIGURATION

A. *Mechanism Overview*

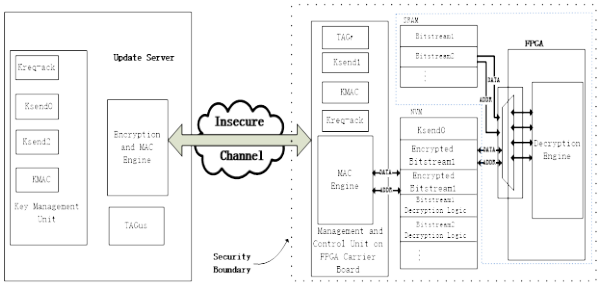

Figure 2. Remote Update Mechanism of Bitstreams

As shown in figure 2:

(1) Remote update configuration end: it isresponsible for generating and sending encrypted bitstream information. It is

composed of the key storage, encryption and MAC engines and bitstream data packets and version identification TAGR. Furthermore, the bitstream data packets include the device ID of FPGA, hardware and software version information, plus the control bits information, data size etc, denoted by B. KB is the corresponding symmetric key of FPGA which is stored in key storage, KB are only known to the client of remote update configuration end and the corresponding FPGA. KMAC is the key which is used to generate the Message Authentication Code (MAC), remote update configuration, the control unit and the FPGA share the key using group encryption mechanism provided by literature [8], users within the group will be able to decrypt encrypted information by KMAC, and users without the group don't have the ability to decrypt the information. TAGR used to identify the bitstream version, which can guarantee the freshness of bitstream version, in order to prevent deception or replay attack in the process of information transmission. Encryption and MAC engine is implemented by the hardware, the bitstream is encrypted and calculate the MAC value, to ensure the confidentiality and integrity of the bitstream.

(2) Control and management unit in SFRCP-end: responsible for receiving information from the remote update server-side, and sent encrypted bitstream to the FPGA. TAGF is a copy of TAGUS on the remote update server-side which is saved in it, and TAGF can only be triggered by the update command to increase of remote update server-side. MAC engine computes MAC value of received encrypted bitstream, and compared with the received MAC value, if match, then determine whether the received encrypted bitstream species, if it is decryption logic bitstream to be stored in the NVM and recorded management and control unit, if it is the encrypted bitstream to be stored in the NVM; otherwise notify a remote update configuration client update failed.

(3) The FPGA: it is divided into two logical function in different time. Concretely, in the decryption function period it mainly includes data input/output and decryption engine. Data input/output unit is implemented by MicroBlaze soft core, which loaded encrypted bitstream into the FPGA from NVM and output decrypted bitstream from decryption engine into SRAM expanded outside the FPGA. Secondary configuration phase is mainly that management and control unit indicates the FPGA to load the decrypted targeted logic from the SRAM, that eventually the FPGA is configurated into the required computational structure.

B. Remote Security Reconfiguration Mechanism

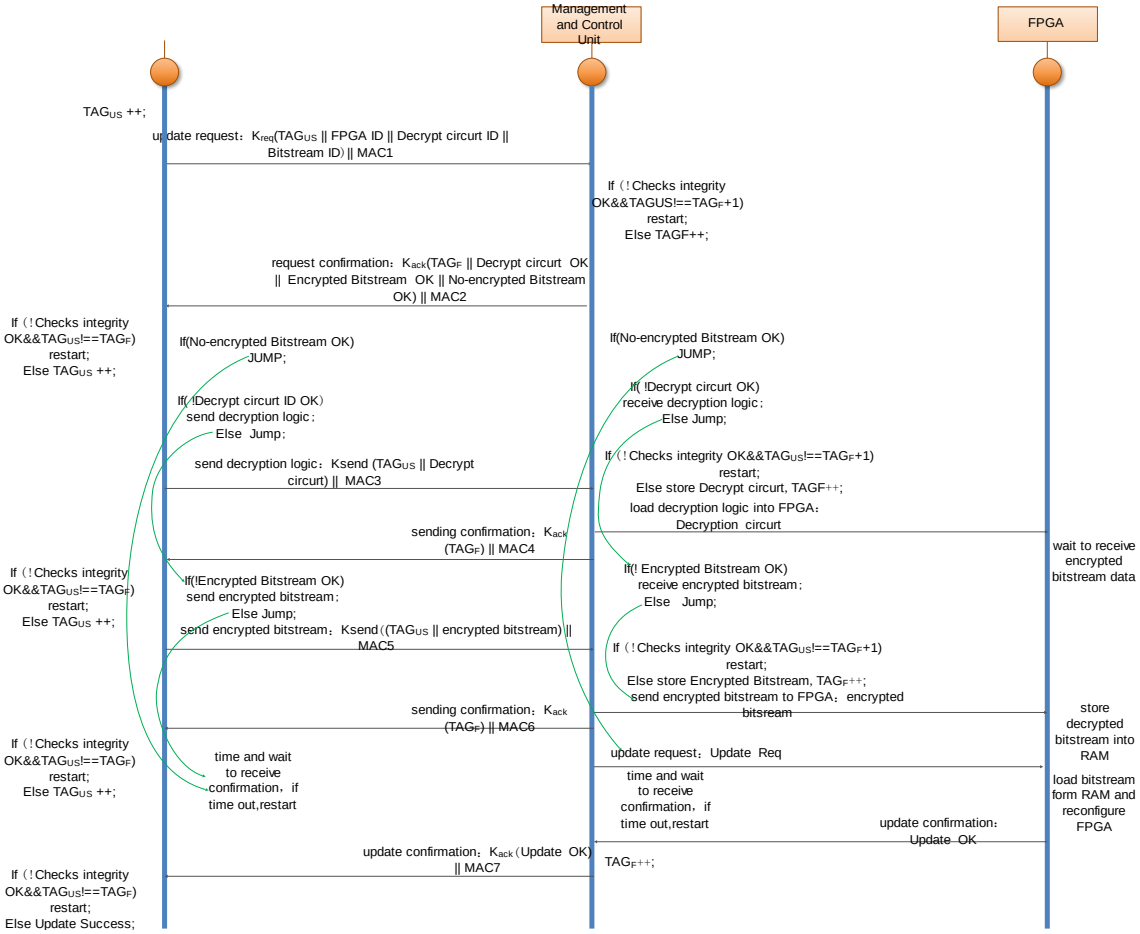

Figure 3. State Machine of the Remote Update Mechanism

Remote update process of the bitstream is showed in Figure 3. Remote update server-side is the only entity that can be updated FPGA configuration and TAG_{US} value. Bitstream can be encrypted using different type of the encryption algorithm, encryption keys also are different, an encryption algorithm can encrypt a plurality of bitstreams. Explanation of symbols used herein are as follows: Ksend(X) represents that message X is encrypted with the symmetric key Ksend; DK(X) represents encrypted message X can be decrypted with the symmetric key K to the message X; $MAC_{KMAC}(X)$ represents computes Message Authentication Code of message X with the key KMAC; | | is on behalf of the interconnected operation of the both messages.

First of all, update server-side state machine operates as follows:

1) to send the update request. First the TAGus is increased by 1, then make TAG_{US} ,FPGA ID to be configured, the ID of the decryption logic and Bitstream ID used to configure the FPGA to connect together, and then encrypt them with the Kreq key; next calculated MAC1 = (MAC_{KMAC} (TAG_{US} || FPGA ID || the Decrypt circurt ID || Bitstream ID), and then send the encrypted information and MAC1 to management and control unit of FPGA bearing plate.

2) to receive update confirmation and process. Update server check for the integrity and freshness of confirmation of update requirement information, if the verification is passed, TAG_{US} increases by 1.

3) to determine whether on the target FPGA there is decrypted bitstream logic to be configured, if it is, state machine jumps to 7; otherwise determine whether on the FPGA carrier card there is decryption logic used to decrypt the target bitstream, if it is, state machine jumps to the 5; otherwise send the corresponding decryption logic to management control unit on FPGA bearing plate, concatenate the TAG_{US} with the Decrypt circuit and encrypt them with Ksend, subsequently calculated the MAC3 = MAC_{KMAC}(TAG_{US} || Decrypt circuit), then the encrypted information and MAC3 are connected together and sent to management and control unit.

4) to receive an acknowledgment message and process. Check for the integrity and freshness of the update confirmation for the sent information, if the check is passed, TAGUS is increased by 1.

5) to determine whether encrypted bitstream logic is already exist on the FPGA carrier board, if it is, state machine jumps to 7; otherwise send the corresponding decryption logic to the management and control unit on FPGA bearing plate, put TAG_{US} and Bitstream end to end and encrypt them with Ksend, next calculate MAC5 = MAC_{KMAC} (TAG_{US} || bitstream), then encrypted information and MAC5 are connected together and send out.

6) to receive an acknowledgment message and process. Check for the integrity and freshness of update confirmaton of the sent information, if the check is passed, TAG_{US} is increased by 1.

7) to time and wait to receive the update results and do the processing. Check for the integrity and freshness of update confirmation of the sent information, if the check is passed, the configuration is successful.

The state machine of SFRCP -end proceed as follows:

1) to wait for receiving the update request, check for the integrity and freshness of the received update request, if th check is passed, state goto 2;

2) according to the FPGA ID, the Decrypt circuit ID and Bitstream ID, the Decryption circuit OK, the Encrypted Bitstream OK and the No-encrypted bitstream OK are generated, which means to the corresponding exist identifier of FPGA decryption logic, encryption bit stream logic and non-encrypted bit stream logic respectively (0 for No, 1 behalf of Yes); TAG_F is connected together with the above-mentioned information and encrypt them with Kack, next calculate MAC2 = (MAC_{KMAC} (TAG_F || the Decrypt circurt OK || Encrypted the Bitstream OK || Non-encrypted the Bitstream OK), the encrypted information and MAC2 are concatenated and sent to update the server-side.

3) If the Non-encrypted bitstream OK=1, in other words the corresponding unencrypted Bitstream belongs to the RAM attached to the FPGA, in this case state machine jump to 7; Secondly, if the FPGA carrier board has the corresponding decryption logic, state machine jump to 5, otherwise program turn to 4.

4) to receives the decryption logic. Check for the integrity and freshness of the received information, if the check is passed, store the decryption logic and increase TAGF by 1; encrypt TAG_F with Kack, next calculate MAC4 = MAC_{KMAC} (Kack (TAG_F)), thus send confirmation message to the update server-side; then configure FPGA chip with decrypt logic.

5) If encrypted Bitstream is exist on the FPGA carrier board namely the Encrypted Bitstream OK = 1, then jump to 7, otherwise go to 6.

6) to receive the encrypted bitstream. Check for the integrity and freshness of the received information, if the check is passed, store encrypted bitstream logic and increase TAG_F by 1; encrypt TAG_F with Kack, next calculate MAC6 = MAC_{KMAC} (Kack (TAG_F)), thus send a confirmation message to the update server-side; and then send the encrypted bitstream logic to the corresponding FPGA to decrypt it.

7) update request is sent to the FPGA.

8) to receive the confirmation of FPGA update message, and then send update confirmation message of the FPGA to the update server-side.

the operation of FPGA side is as follows: If the decryption logic is configured into the FPGA, wait to receive encrypted bitstream logic and decrypt the received encrypted bitstream logic, next output to the RAM which is subsidiary of the FPGA; subsequently receive notification of the FPGA update, and then load the corresponding bitstream logic from the RAM to reconfigure the FPGA, after the success of configuration make a response to the management and control unit of the bearing plate.

IV. THE SECURITY ANALYSIS

In this paper the secure update machinism of the remote bitstream uses symmetric encryption to ensure the

confidentiality of the bitstream, and exploits message authentication code MAC and version identifier TAG to ensure the integrity and freshness of each communication information. Also a variety of encryption algorithms is used to encrypt the bitstream for enhancing the security of the encrypted information, because this will increase the difficulty and expense of the attack. Security Solutions of this article assume that the underlying key, encryption module and the MAC function is secure. Therefore, the safety analysis for the bitstream focus on preventing eavesdropping, replay attacks and tampering between the remote update server side and SFRCP end.

In the bitstream communication process of the both ends which interact remotely, adversary can masquerade as remote update configuration end or SFRCP end of the bitstream to intercept. The eavesdropping means that adversary gain the communication information by listening communication network, but the content of our communications are encrypted, the adversary effectively use the obtained communication information is difficult; tampering attack occurs when an attacker attempt to modify the original encrypted bitstream information and then send this pathological information to the SFRCP end, however, in our security scheme we use the MAC to verify the information to ensure the integrity of the bitstream. Once the adversaries modify the information, those changes can be found by calculating and matching the MAC value.

Replay attack amounts to that rival monitor and record the bitstream update information, and then send the expired bitstream update information at a time, the dangers of such attacks on the system could causes the system performance degrading and generating mistake of the function. We utilize the version identification TAG to ensure the freshness of the transmission bit stream to prevent replay attacks in the MAC calculation process[3]. The packets of security update protocol for the communication interaction contain different TAG value, the TAG identifier is encrypted and encapsulated in the package of the MAC value，in both ends of the security communication protocol MAC value is calculated and decrypted the TAG value, next through comparing the decrypted TAG with the stored TAG we can justify the validity of the received data, and analyze the freshness of the bitstream in accordance with the result of the comparation.

V. CONCLUSIONS

Reconfigurable computing platform based on Sram FPGA(SFRCP) is an important high-performance computing platform, we propose a secure scheme for the remote update of bitstream against the security threats which faces with such as eavesdropping, tampering and replay attacks and so on for the SFRCP platform. The scheme will reconfigure the FPGA component with the bitstream information stored in the the NVM in the management and control unit, the management and control unit is wholly in charge of the management of bitstreams and the operation of decryption to reduce the complexity of the remote bitstream update. The bitstream data is encrypted and the MAC value is calculated on the remote update server-side in this paper, and then the information and instruction of update is sent via Ethernet to the SFRCP platform. The SFRCP Platform first verifies the integrity and freshness of the received information by the management and control unit, and then the encrypted bitstream and MAC values are stored into NVM. Finally, when the FPGA is powered up, it requests the control unit and to obtain the bitstream, thus the targeted FPGA is reconfigured in order to safely complete the update of the remote bitstream. The analysis shows that the proposed machinism is able to ensure that the integrity and confidentiality of the update bitstream and prevent replay attacks.

REFERENCES

[1] X. Feng, R. Ge, et al. Power and Energy Profiling of Scientific Applications on Distributed Systems[C]. 19th International Parallel and Distributed Processing Symposium (IPDPS 05), Denver, CO, 2005.

[2] Yan Li. The Secutiy Research of FPGA Configuration Bitstream [D]. Harbin: Harbin Engineering University, 2008.

[3] W. Adi, R. Ernst, et al. Vlsi design exchange with intellectual property protection in fpga environment using both secret and public - key crytography[C]. Emerging VLSI Technologies and Architecture, 2006. IEEE Computer Society Annual Sympoium, 2 - 3 March 2006.

[4] J.Castilo, P.Huerta, and J.Martinez. Secure ip downloading for sram fpga[J]. Microprocessors and Microsystems,2007, 31: 77 - 86.

[5] S. Drimer and M. G. Kuhn. A protocol for secure remote updates of fpga configuration[C]. Proceedings of the 15th International Workshop on Reconfigurable Computing: Architecture, Tools and Applications. Berlin, Heidelberg: Springer - Verlag, 50 - 61, 2009.

[6] F. Devic, L. Torres, and B. Badrignans, Secure Protocol Implementation for Remote Bitstream Update Preventing Replay Attacks on FPGA[C]. In Proceeding of the 20th International Conference on Field Programmable Logic and Applications, 179 - 182. 2010.

[7] Wang Qin, Sun Fu-mig, etc. Overview of FPGA Design Security[J]. Journal of Chinese Computer Systems, 2010, 31(7): 1333 - 13337.

[8] GUANG - HUEI C and WEN - TSUEN C. Secure broadcasting using the secure lock[J]. Transactions on Software Engineering, 1989, 15(8): 929 - 934.

[9] Kun - Wah Yip and Tung - Sang Ng. Partial - Encryption Techique for Intellectual Property Protection of FPGA - Based Products[J]. IEEE Transactions on Consumer Electronics, 2000, 46(1):183 - 190.

[10] Braeken A, Genoe J, et al. Secure Remote Reconfiguration of an FPGA - based Embedded System[C]. in 6th International Workshop on Reconfigurable Communication - centric Systems - on - Chip(ReCoSoC), 1 - 6, June 2011.