Secure and Reliable XOR Arbiter PUF Design: An Experimental Study based on 1 Trillion Challenge Response Pair Measurements

Chen Zhou, Keshab K. Parhi, Chris H. Kim
Department of Electrical and Computer Engineering
University of Minnesota, Minneapolis, MN, USA, 55455
{zhoux825, parhi, chriskim}@umn.edu

ABSTRACT

This paper shows that performing an XOR operation between the outputs of parallel arbiter PUFs generates a more secure output at the expense of reduced stability. In this work, we evaluate the security and stability of XOR PUFs using 1,000,000 randomly chosen challenges, applied to 10 custom-designed PUF chips, tested for 100,000 cycles per challenge, under different voltage and temperature conditions. Based on extensive hardware data, we propose a practical method for selecting challenges that will produce stable responses. A linear regression approach based on soft responses collected during enrollment phase was used to build accurate models for each individual arbiter PUF. Hardware data from fabricated chips verify that the approach is highly effective.

CCS Concepts

•Security and privacy → Security in hardware •Hardware → Application-specific VLSI designs.

Keywords

XOR PUF; response stability; challenge selection; modeling attack; silicon data; VDD and temperature variation

1. INTRODUCTION

Multiplexer (MUX) based arbiter Physical Unclonable Function (PUF) is a promising candidate for hardware authentication applications as the total number of challenge response pairs (CRPs) is an exponential function of the number of delay stages. Security, stability, storage requirement, test complexity, and authentication time are important considerations when designing MUX arbiter PUFs. In particular, ensuring a stable and consistent response across a wide range of temperature, voltage, and aging conditions is a critical challenge. When the arbiter compares two path delays with a small delay difference, random thermal noise can cause intermittent errors. To overcome this issues, an authentication strategy based on only stable CRPs was recently proposed and verified through silicon data [1]. Here, stable CRPs were selected based on "soft response" measurements, which indicate the degree to which a response is stable. Soft responses were measured using an on-chip counter that repeatedly samples the response for a given challenge to generate the average response value.

Vulnerability to machine learning attacks is another serious concern in MUX arbiter PUFs. That is, a hacker can easily mimic the PUF by building a software model based on a small number of CRP measurements. Several recent works have shown that the delay difference of each MUX PUF stage can be predicted accurately using a linear additive model [2-5]. These works show that delay parameters can be extracted relatively easily using machine learning techniques, such as logistic regression. Other related works have proposed storing delay parameters rather than an exhaustive CRP dataset, to reduce the storage and computation requirements of the server [4-7].

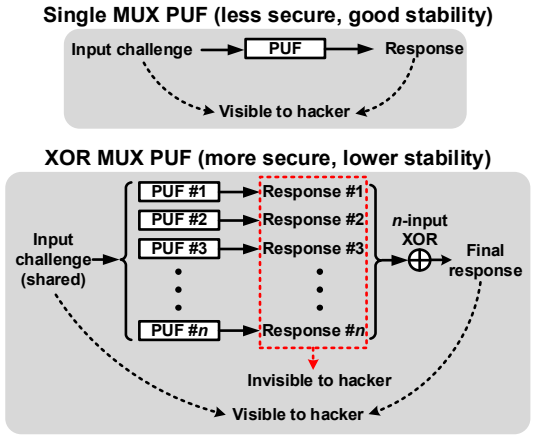

Figure 1. Standard MUX PUF (top) and XOR MUX PUF (bottom). The XOR PUF is more secure than a standard PUF, but produces more unstable responses.

To realize a PUF structure that is more resistant to modeling attacks, an XOR arbiter PUF was proposed in [8]. The basic principle is described in Fig. 1 where a bank of PUFs presented with a common challenge generates *n* output responses, which are then XOR-ed together to produce the final 1-bit response. The premise for XOR PUFs is that estimating the responses of individual PUFs based on the final XOR output is very difficult and time-consuming. This is why XOR PUFs are believed to be more secure than standard MUX PUFs. As discussed in [3], the security and stability of an XOR PUF strongly depend on the number of PUFs used for the XOR operation. The training time of a MUX PUF model increases exponentially with the number of parallel PUFs, which is highly desirable from a security standpoint. However, the percentage of unstable responses also increases exponentially, which means that the higher security of XOR PUFs comes at a price of lower stability. Since Hamming distance based PUF authentication policies can only tolerate a certain amount of

DAC '17, June 18-22, 2017, Austin, TX, USA

DOI: http://dx.doi.org/10.1145/3061639.3062315

noisy responses, the number of PUFs in an XOR PUF is typically kept to be 6 or less, as discussed in previous works [3, 6-7, 9]. However, recent XOR PUF modeling works have shown that it is not too difficult to attack a XOR PUF with up to 6 internal PUFs [3, 6-7, 9]. To improve the security of XOR PUFs, several approaches utilizing pre-extracted delay parameters have been introduced. For example, a noise bifurcation architecture allows only a fraction of the randomly chosen responses to be sent back to the server for the actual authentication [6]. The CRPs that are available to the hacker are randomly disturbed, thus making modeling attacks more difficult. However, the authentication criterion must be relaxed considerably in this case, requiring a higher number of CRPs for a reliable authentication. A lockdown technique in [7] makes CRPs only accessible with the server's permission, and thus prevents the hacker from obtaining enough CRPs. However, this strategy requires complicated system level support.

In this work, we first evaluate the resiliency of XOR PUF to modeling attacks using 1 trillion CRPs obtained from multiple 32nm test chips. Measured data suggests that more than 10 individual PUFs are needed for an XOR PUF to be considered secure. To find challenges that produce a stable XOR response, delay parameters are extracted from each individual PUF during enrollment phase. On-chip fuses allow one-time access to the individual PUFs during the enrollment phase, but block access during authentication phase. Experimental results show that CRPs selected based on our PUF model provide excellent stability under different voltage and temperature, making XOR PUFs attractive for authentication application.

2. SECURITY AND STABILITY EVALUATION OF XOR PUF

In this section, we evaluate the stability of a single MUX arbiter PUF and then assess the security and stability of an XOR PUF with different number of parallel PUFs. Hardware data was collected from custom-designed 32nm PUF test chips.

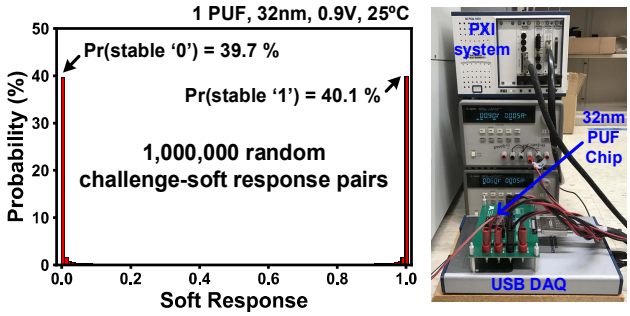

Figure 2. Soft response (i.e., Pr(response='1')) distribution measured from 32nm PUF chips, by applying 1,000,000 random challenges. The soft responses were obtained by applying the same challenge 100,000 times and using on-chip counters to compute the average response value. The soft response has a range from 0.00 to 1.00 with a bin size of 0.01.

2.1 Soft Response Measurements

To understand the performance of a standard linear MUX delay arbiter PUF, we fabricated custom PUF chips in a 32nm process. Soft responses were collected by evaluating the PUF 100,000 times for each challenge. We tested 10 PUF chips with 1,000,000 random challenges. The soft response distribution of a single MUX arbiter PUF is shown in Fig. 2. The bin size of the histogram is 0.01. The soft response value ranges from 0.00 to 1.00 depending on the internal path delay difference and random noise. Of the 1,000,000 challenges we applied, 39.7% produces a soft response of 0.00 which represents paths with a large negative delay difference, while 40.1% produces a soft response of 1.00 representing paths with a large positive delay difference.

2.2 XOR PUF Stability Evaluation

The final response of an XOR PUF is generated by taking the XOR between the outputs of parallel PUFs. Soft responses that are stable enough for a single MUX PUF (e.g., 0.05 or 0.95) may not be sufficient for an n-input XOR PUF with a large n value [1]. This is because instability in any individual MUX PUF will cause the final XOR output to become unstable. To guarantee the stability of XOR response in authentication, only the challenges that produce 100% stable responses on all PUFs can be used. These are the challenges with soft response = 0.00 or 1.00. The reason why we suggest only using the challenges known to generate 100% stable responses in authentication, while avoiding slightly unstable responses, is because if all individual responses are slightly unstable, the final XOR results could be highly unstable with a large n value. However, if soft responses can be collected for the final XOR PUF responses and reasonable thresholds are applied, marginally stable responses could also be salvaged for use in authentication. In this work, we only focus on responses that are 100% stable since the authentication process is simpler and more straight-forward. For instance, sampling the XOR output once is sufficient since the individual responses will always generate the same output. If we limit our choices to CRPs that are 100% stable for all individual PUFs, then the number of useable CRPs will drop exponentially with n. As shown in Fig. 2, the percentage of stable CRPs for a single standard MUX PUF is roughly 80%. If the individual PUFs are uncorrelated, then the percentage of stable CRPs for an n-input XOR PUF is roughly 0.8^n. This estimation matches our silicon test data shown in Fig. 3. It shows that for a 10-input XOR PUF, the percentage of challenges producing a stable response is only 10.9%. Previous works did not focus on XOR PUFs with a large n value such as 10, and as a consequence, the stability issue of XOR PUFs was overlooked. In this work, we address both the security and stability aspects of XOR PUFs with up to 10 parallel MUX PUFs.

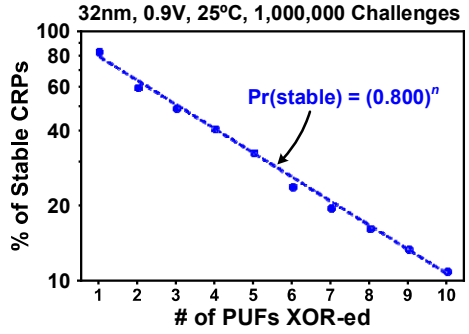

Figure 3. The percentage of stable CRPs versus the number of PUFs in an XOR PUF. For a 10-input XOR PUF, only 10.9% of the measured CRPs are stable.

2.3 XOR PUF Security Evaluation

XOR PUFs are known to be more secure than a single MUX PUF; however, they are not completely immune to advanced machine learning techniques. In Fig. 4, we show modeling attack results on an n-input XOR PUF using our test chip data. We collected responses from each individual MUX PUF and performed an XOR operation. In our experiment, we found that models trained with only stable CRPs are more accurate in predicting test set responses.

In other words, unstable XOR PUF CRPs have the tendency to mislead the model training. Therefore, all the XOR PUF modeling attack experiments in our work are based on 100% stable XOR PUF responses, in which case all individual PUFs produce 100% stable responses. Similar to the training set case, we employed only the 100% stable CRPs for the test set. This decision is based on our authentication scenario assumption that only stable CRPs will be used in actual authentication. The model prediction accuracy for the unstable CRPs doesn't matter to us since those CRPs will be discarded anyway. In our experiments, we measured 1,000,000 challenges for each individual PUF. 90% of the 1,000,000 challenges were used as the training set while the remaining 10% was used as the test set. Since we use only the stable CRPs for both training and testing, the actual set sizes are less than 900,000 and 100,000, respectively, and vary depending on the n value. For instance, the maximum training set for a XOR PUF with n individual PUFs is roughly $900{,}000*0.800^n$, while the related test set size is $100{,}000*0.607^n$. 80.0% is the percentage of CRPs that are stable according to measurements for a single MUX PUF, while 60.7% is the percentage that are predicted to be stable by the linear additive model. The training was performed using a multi-layer perceptron classifier model. We built a 3-layer neural network comprising of 35 (first layer), 25 (second layer) and 25 (third layer) nodes. This configuration provided a good balance between model accuracy and training time. Please note that a larger network always leads to longer training time, but doesn't always results in higher accuracy. Transformed challenge vectors were applied as training inputs, which is a widely used method for linear MUX arbiter PUF modeling [1-3]. One-bit XOR PUF responses were used as training targets. Soft responses were not applied since only the 100% stable XOR PUF responses were used in this work. The optimization algorithm is the Limited-memory Broyden–Fletcher–Goldfarb–Shanno (BFGS) provided in the scikit-learn toolkit [10]. The training was performed on a desktop computer with an Intel i7-4770 CPU and a 16 GB RAM. As shown in Fig. 4, for n<10, the prediction accuracy of the trained model can reach 90% using less than 100,000 CRPs. These results suggest that the number of parallel PUFs should be at least 10 for an XOR PUF to be resistant to machine learning attacks. We found that the training time is related to the number of CRPs but only a weak function of n. The average training speed is 0.395ms per CRP, which is shorter compared to previous works [3, 6-7, 9]. The training accuracy and efficiency may be related to the number of MUX stages inside the MUX PUF. Each MUX PUF implemented in the 32nm test chip has 32 MUX stages.

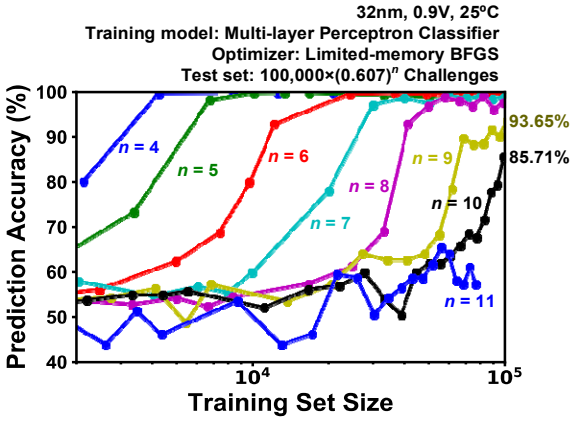

Figure 4. Prediction accuracy of artificial neural network based model as a function of training set size and the number of individual PUFs in XOR PUF. The data suggests that more than 10 PUFs are needed for the XOR PUF to be considered secure.

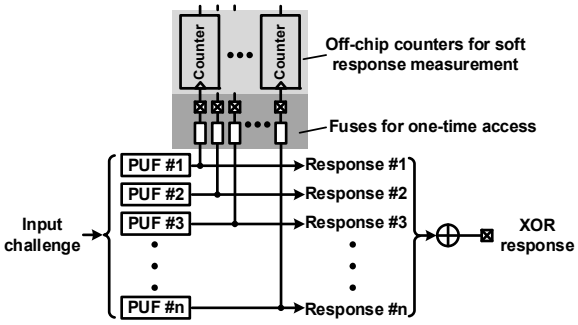

Figure 5. Proposed model-assisted XOR PUF design with one-time PUF access via fuses and off-chip soft response collection.

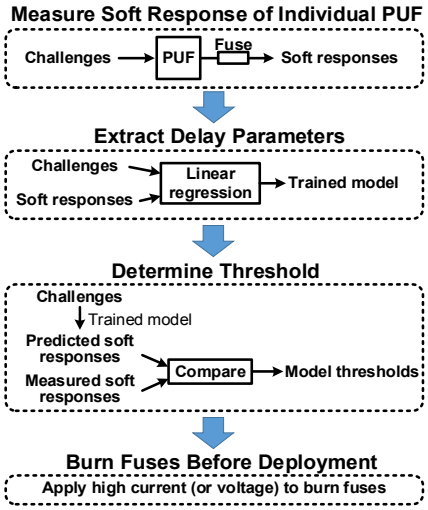

Figure 6. Enrollment phase of the proposed model-assisted XOR PUF.

3. PROPOSED XOR PUF OPERATION

Next, we present a new authentication strategy targeted for wide XOR PUFs where highly stable CPRs are selected using a software model. Figs. 5 and 6 show the XOR PUF hardware and enrollment procedure, respectively. On-chip fuses and off-chip counters are employed so that an authorized tester can access the soft response of each individual PUF during enrollment phase. Alternatively, on-chip counters can be used to efficiently measure the soft responses without having to scan out every single response bit [1]. Since response values are averaged over thousands of cycles, soft responses are less noisy compared to hard responses, and therefore allows a more accurate estimation of the delay parameters. The delay parameters are extracted for each PUF based on the measured soft responses, and stored in the server database [4, 6-7]. We also introduce a procedure for determining the thresholding criterion. The aim is to find the thresholding levels for classifying the CRPs into three prediction categories: stable '0', stable '1' and unstable. This can be accomplished by comparing the model predicted results with the soft response measurements. Further details of the thresholding method will be given in Sections 4 and 5. Once the enrollment is complete, the fuses are blown by applying a high current or voltage. This makes it impossible for the hacker to access the individual PUF responses [11]. Fig. 7 shows the proposed authentication flow. Unlike traditional approaches where randomly chosen challenges are applied, only the ones that are predicted to produce stable responses in all individual PUFs are used in

authentication. This challenge selection procedure is performed by the server using the delay parameters and model thresholds found during enrollment phase. The returned responses are compared with the responses predicted by the server for final authentication.

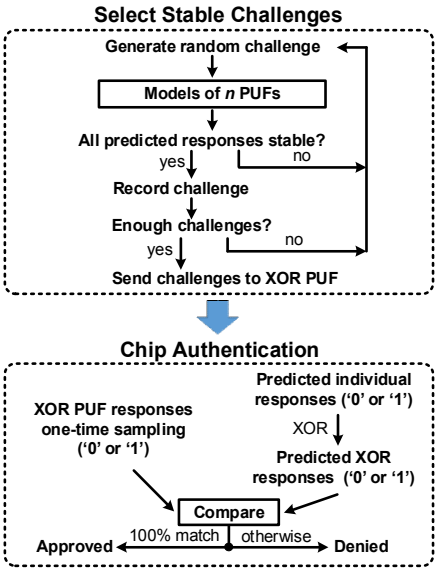

Figure 7. Authentication phase of the proposed model-assisted XOR PUF.

Since the CRPs used in our authentication scheme are extremely stable, a very stringent approval criterion can be imposed without causing errors. For instance, the server may grant access only when the client responses and server predicted responses match perfectly (i.e., zero Hamming distance). This is an important contribution of our work that is different from previously published works. Our proposed strategy improves the overall security of the system by imposing a much stringent authentication criterion. Notice that even though the idea of applying only stable CRPs in authentication has already been introduced in [1], their CRPs selection scheme was solely based on hardware measurements. This strategy is acceptable for a single MUX PUF, however, it is less efficient for XOR PUFs with a large n where most tested CRPs are discarded due to poor stability. On the other hand, the proposed CRP selection scheme only requires a small number of CRPs for delay parameter extraction and can predict the stability of CRPs that haven't been tested during enrollment phase.

4. LINEAR MODEL TRAINING

We adopted the linear additive delay model to estimate the delay parameters [1-3]. However, there are two main differences in our work compared to others: (1) we use the linear regression algorithm, rather than logistic regression; (2) we apply thresholds to classify the prediction results into three categories (i.e., stable '0', stable '1' and unstable) rather than two categories used in previous works. The reason for the first difference is quite simple: we obtained soft responses that are fractional numbers, rather than binary numbers. As for the second difference, classifying responses into three categories helps select only the highly stable CRPs. The traditional two-category approach decides the binary response by simply applying a threshold of 0.5 which is prone to flipping errors.

The model thresholds are determined by comparing the measured soft responses with the model predicted soft responses, as shown in Fig. 8. The soft responses vary from 0.00 to 1.00, while the model prediction soft responses have a wider range but are still centered around 0.5. As we will see in Section 5, the wider range of the predicted soft responses provides additional stability information for each challenge. Only soft responses that belong to the first (i.e., 0.00) and last (i.e. 1.00) bins of the histogram are qualified as stable responses in measurement. Model predicted CRPs are divided into three non-overlapping categories as shown in Fig. 8 (upper). The zoomed-in view in Fig. 8 (lower) illustrates the threshold levels for '0' and '1'. The lowest predicted soft response to result in a measured soft response greater than 0.00 is defined as $threshold('0')$. Similarly, the highest predicted soft response to result in a measured soft response less than 1.00 is defined as $threshold('1')$. From the figures, we can see that not all stable CRPs in measurement are predicted as stable CRPs using the model. It's worth pointing out that the predicted soft response is a measure of the internal path delay difference. A stronger bias in the soft response suggests a larger positive or negative delay difference. CRPs that are stable in measurement but are unstable in the model are suspected as marginally stable, and are likely to become unstable with voltage and temperature variation. To be on the safer side, it's better to discard them. Since we discard marginally stable CRPs, there are fewer predicated stable CRPs than measured stable CRPs. The ability to discriminate between absolutely stable CRPs and marginally stable CRPs is a unique advantage of the proposed model-based CRP selection scheme.

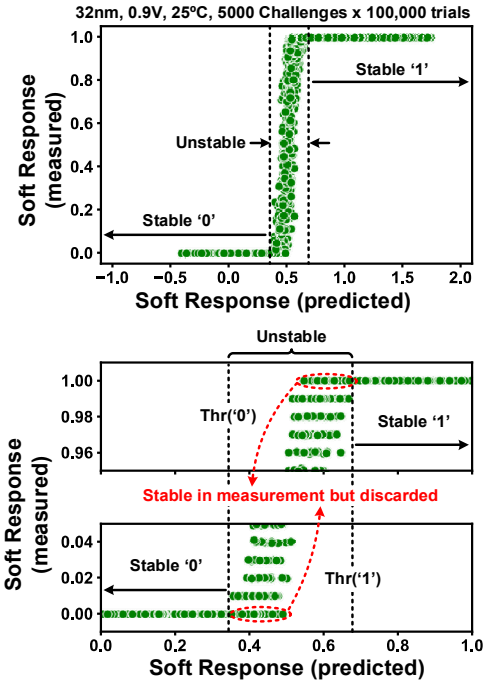

Figure 8. Comparison between measured soft response and model predicted soft response: full plot (above); zoomed-in plot (below).

5. THRESHOLD LEVEL ADJUSTMENT

The threshold values found in the previous section can be directly applied during the authentication process. However, there remains a risk that some CRPs predicted as stable may become unstable at different supply voltages or temperatures. Furthermore, these threshold values may not be stringent enough for CRPs not used for the model extraction. To ensure that the thresholds are reliable for any test set and any test condition, we adjust the thresholds of the training set to more stringent values. This way, the model selected CRPs are stable even for CRPs that were never measured. The simplest way to decide the new threshold levels is to arbitrarily set

them to extremely stringent values; for example, 0.0 for stable '0' and 1.0 for stable '1'. As can be seen from Fig. 8, thresholds of 0.0 and 1.0 are far away from the unstable region in the center. The path delay differences corresponding to these challenges are quite large guaranteeing a stable response. However, we also observe that for such stringent thresholds, a large number of stable CRPs could be discarded, resulting in fewer usable CRPs for authentication. To maximize the number of usable CRPs without compromising the response stability, we propose threshold scaling factors β_0 and β_1. That is, the thresholds used for authentication is scaled down by a factor of β_0 for '0' and scaled up by a factor of β_1 for '1'. The β_0 and β_1 values can be determined based on trial-and-error or based on simple heuristics.

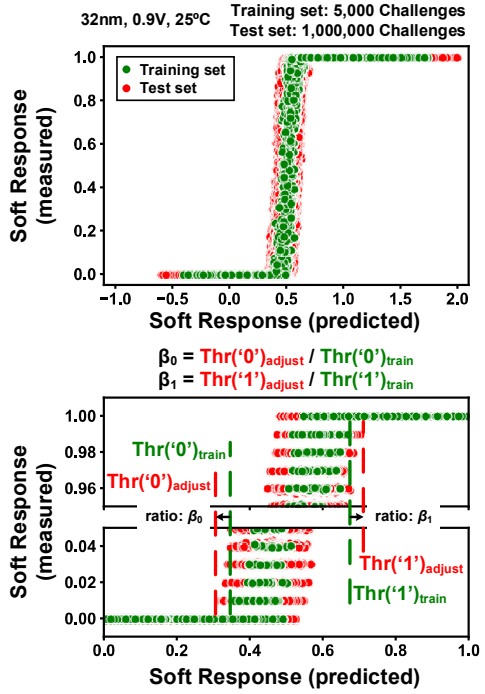

Figure 9. Threshold levels can be made more stringent by scaling down the threshold for '0' by β_0 (<1) and scaling up the threshold for '1' by β_1 (>1) during actual chip authentication. This figure shows a simple guideline for determining scaling factors β_0 and β_1.

5.1 Thresholds under Nominal Condition

Fig. 9 shows a guideline for determining β_0 and β_1. The training set comprises of 5,000 CRPs while the test set has 1,000,000 CRPs, both under a nominal voltage and temperature condition. There remains a small chance that $threshold('0')$ and $threshold('1')$ obtained from the training set may render incorrect responses for the test set. To mitigate this issues, we adjust $threshold('0')$ to $\beta_0 * threshold('0')$ and $threshold('1')$ to $\beta_1 * threshold('1')$ for the test set evaluation. β_0 and β_1 are both initially set to 1.00. We gradually decrease β_0 and increase β_1, until all unstable responses are filtered out. We could use different β_0 and β_1 values for different PUFs. However, in realistic operation, it is more convenient to use the same beta values for all PUF chips. We can decide the common values based on a small sampling of PUF chips and apply them to all other PUFs. The β_0 and β_1 values extracted from 10 different PUFs vary between 0.74~0.93 and 1.04~1.08, respectively. We chose the most conservative values; i.e., β_0=0.74 and β_1=1.08.

In previous discussion, we kept the training set size to be 5,000 CRPs. It is worth studying the effect of training set size on the response stability. During several modeling attempts, we found that the prediction accuracy of linear additive delay model strongly depends on the number of CRPs used for training, and that a more accurate model can provide more stable CRPs for authentication. To this end, we trained the model while varying the training set size from 500 to 10,000 CRPs and checked the number of stable CRPs the model can find. As shown in Fig. 10. The percentage of stable CRPs predicted by the model saturates at ~60% after the threshold adjustment scheme discussed above, compared to ~80% in measurement. Considering the tradeoff between testing cost and model accuracy, we chose 5,000 CRPs as the train set. The training time was 4.3*ms* on our desktop computer.

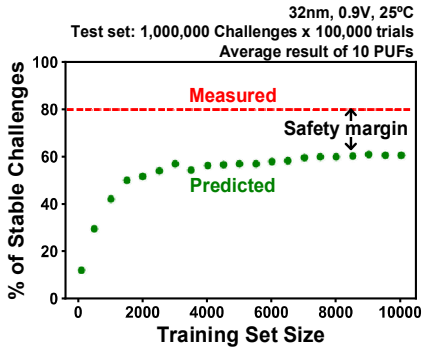

Figure 10. The probability of stable CRPs (both measured and predicted) versus training set size.

5.2 Thresholds under Voltage and Temperature Variation

Most of the previous PUF papers investigated stability under nominal conditions. Only few works have explored the response stability under different voltages and temperatures. This is primarily due to the lack of massive test data from a real chip which is the only way to capture actual voltage and temperature effects. In our work, we tested the fabricated PUFs under different voltage and temperature conditions with 1,000,000 random challenges. By adopting the threshold adjustment scheme discussed before, the trained model can also predict stable responses under various test conditions correctly.

Fig. 11 displays the soft response distribution of both the training set and test set. Similar to Fig. 9, we used a training set of 5,000 challenges under a nominal condition; i.e., 0.9V and 25°C. 1,000,000 test challenges were applied and responses were measured at 0.8V, 0.9V and 1.0V while varying the temperature from 0°C, 25°C, to 60°C. Consequently, Fig. 11 captures 1,000,000 CRP measurements under 9 different test condition. Compared to Fig. 9, the test set distribution in Fig. 11 is much wider due to the test condition variation. However, since the unstable CRPs remain concentrated in the middle region, the threshold adjustment scheme in Section 5 is still effective. Compared to the nominal case, β_0 and β_1 are adjusted from 0.74→0.44 and 1.08→1.33, respectively, to account for voltage and temperature effects.

The ability to mitigate voltage and temperature effects is another advantage of our proposed linear model based challenge selection scheme. The traditional measurement-based approach [1] requires testing the PUF under different conditions to select CRPs that are immune to test condition variation. However, this requires the supply voltage and temperature to be varied during chip testing which is time-consuming and tedious. Unlike the traditional

approach, our model-based approach simply applies more stringent thresholds during authentication and hence does not require measurements at different operating conditions.

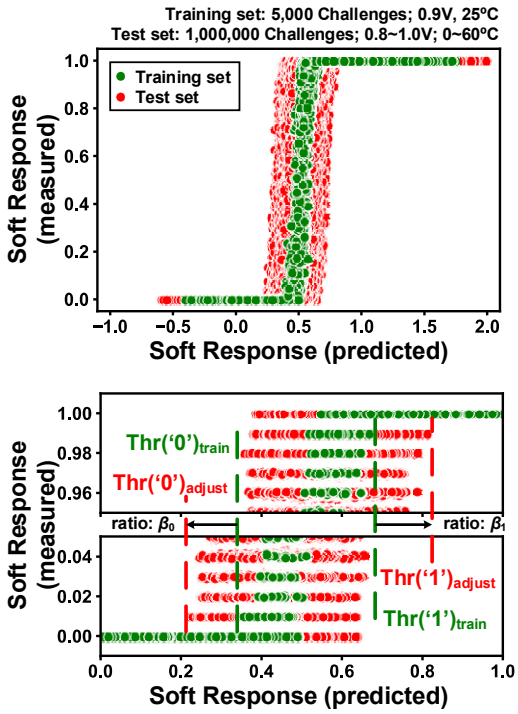

Figure 11. Threshold level adjustment scheme is applied for responses under different test conditions. More stringent β_0 and β_1 values are needed to overcome voltage and temperature effects.

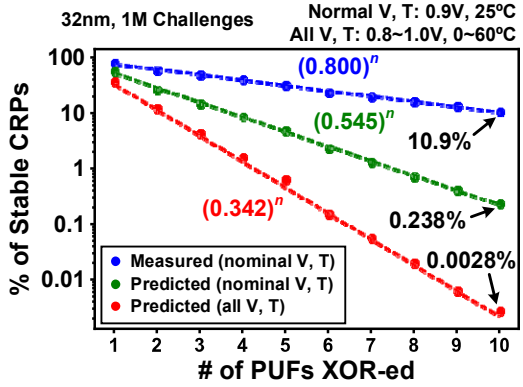

Figure 12. The probability of stable CRPs versus the number of individual PUFs in XOR PUF under different voltage and temperature conditions

As we discussed above, the number of predicted stable CRPs is strongly related to the threshold levels. More stringent thresholds result in fewer available CRPs. The percentage of stable CRPs (measured and predicted) versus the number of PUFs in a XOR PUF is shown in Fig. 12. Thresholds with and without consideration of test condition variation were both studied. All curves follow an exponential trend, suggesting a negligible correlation between the individual PUFs. If 10 parallel PUFs are used in XOR operation, the percentage of stable CRPs predicted by the model is just 0.238% under a nominal condition. This percentage drops to 0.0028% with voltage and temperature variation. Although this number is much lower than that in measurement (i.e., 10.9%), these are the challenges that will remain stable under extreme voltage and temperature variations. Besides, MUX PUFs have a huge CRP space to work with. For instance, a 64-stage MUX PUF has $2^{64} = 1.8 \times 10^{19}$possible challenges, so 0.0028% of this number ($= 0.0028\% \times 2^{64} = 5.17 \times 10^{14}$) is still sufficiently large for practical authentication applications.

6. CONCLUSION

In this work, we evaluated the security and stability of XOR PUF using 1 trillion CRP data measured from 32nm test chips. Experimental data shows that a secure XOR PUF should include no less than 10 parallel PUFs. To enable reliable XOR PUF authentication, we propose a linear regression model based challenge selection strategy. We fully evaluate this strategy using silicon data and suggest practical ways of implementation. Finally, we show experimental data confirming that the challenges selected based on our approach can produce highly stable responses under different voltages and temperatures.

ACKNOWLEDGEMENTS This research was supported in part by the National Science Foundation under grant number CNS-1441639 and the Semiconductor Research Corporation under contract number 2014- TS-2560. The authors thank Dr. Amitabh Das at Intel for providing invaluable technical feedback.

REFERENCES

[1] Zhou, C., Satapathy, S., Lao, Y., Parhi, K., and Kim, C.H. 2016. Soft Response Generation and Thresholding Strategies for Linear and Feedforward MUX based PUFs. *International Symposium on Low Power Electronics and Design.*

[2] Delvaux, J. and Verbauwhede, I. 2013. Side channel modeling attacks on 65nm arbiter PUFs exploiting CMOS device noise. *IEEE International Symposium on Hardware-Oriented Security and Trust*, 137-142.

[3] Rührmair, U., et al., PUF Modeling Attacks on Simulated and Silicon Data. 2013. *IEEE Transactions on Information Forensics and Security*, 1876-1891.

[4] Xu, X., Burleson, W. and Holcomb, D. E. 2016. Using Statistical Models to Improve the Reliability of Delay-Based PUFs. *IEEE Computer Society Annual Symposium on VLSI*, 547-552.

[5] Avvaru, S. V. S., Zhou, C., Satapathy, S., Lao, Y., Kim, C. H. and Parhi, K. K. 2016. Estimating delay differences of arbiter PUFs using silicon data. *Design, Automation & Test in Europe Conference & Exhibition*, 543-546.

[6] Yu, M. D., M'Raïhi, D., Verbauwhede, I. and Devadas, S. 2014. A noise bifurcation architecture for linear additive physical functions. *Hardware-Oriented Security and Trust*, 124-129.

[7] Yu, M. D., Hiller, M., Delvaux, J., Sowell, R., Devadas, S. and Verbauwhede, I. 2016. A Lockdown Technique to Prevent Machine Learning on PUFs for Lightweight Authentication. *IEEE Transactions on Multi-Scale Computing Systems*, 146-159.

[8] Suh, G. E. and Devadas, S. 2007. Physical Unclonable Functions for Device Authentication and Secret Key Generation. *Design Automation Conference*, 9-14.

[9] Becker, Georg T. 2015. The gap between promise and reality: on the insecurity of XOR arbiter PUFs. *International Workshop on Cryptographic Hardware and Embedded Systems*, 535-555.

[10] Pedregosa, F., Varoquaux, G., Gramfort, A., et al. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 2825-2830.

[11] Rostami, M., Majzoobi, M., Koushanfar, F., Wallach, D. S. and Devadas, S. 2014. Robust and Reverse-Engineering Resilient PUF Authentication and Key-Exchange by Substring Matching. *IEEE Transactions on Emerging Topics in Computing*, 37-49.