

Tuning the Winternitz Hash-based Digital Signature Scheme

Lucas Perin Gustavo Zambonin Douglas Martins
Ricardo Custódio Jean Martina

Laboratório de Segurança em Computação
Universidade Federal de Santa Catarina

`lucas.perin@posgrad.ufsc.br`

ISCC 2018
Natal, Brazil
June 27, 2018

Outline

Winternitz signature scheme

Tuning Winternitz

Final remarks



Winternitz signature scheme

Winternitz signature scheme

Classical Winternitz

Discussion

Related works

Tuning Winternitz

Final remarks



Winternitz

Key generation

Let H be some hash function and n the block size of H .
Choose w such that:

$$t_1 = \left\lceil \frac{n}{w} \right\rceil, \quad t_2 = \left\lceil \frac{\lfloor \log_2 t_1 \rfloor + 1 + w}{w} \right\rceil, \quad t = t_1 + t_2.$$



Winternitz

Key generation

Let H be some hash function and n the block size of H .
Choose w such that:

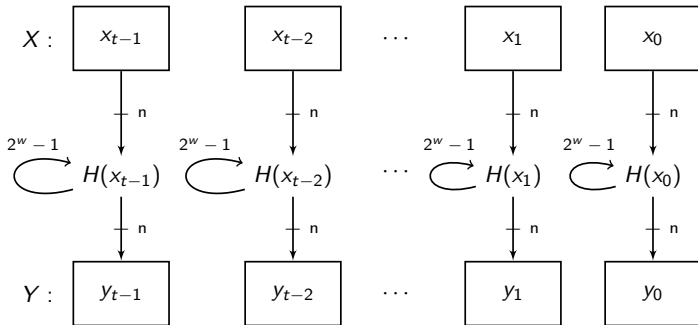
$$t_1 = \left\lceil \frac{n}{w} \right\rceil, \quad t_2 = \left\lceil \frac{\lfloor \log_2 t_1 \rfloor + 1 + w}{w} \right\rceil, \quad t = t_1 + t_2.$$

For SHA-256 and $w = 4$, we have $t_1 = 64$, $t_2 = 3$ and $t = 67$.

Pick a random $X = (x_{t-1}, \dots, x_1, x_0) \in \{0, 1\}^{n,t}$,

Pick a random $X = (x_{t-1}, \dots, x_1, x_0) \in \{0, 1\}^{n,t}$,

and let $Y = (H^{2^w-1}(x_{t-1}), \dots, H^{2^w-1}(x_0))$.





Winternitz

Signature generation

Message M and hash function H such that:

$$H(M) = d = \underbrace{b_{t-1}}_w \parallel \cdots \parallel \underbrace{b_{t-t_1}}_w \quad \text{where } 0 \leq \text{int}(b_i) < 2^w - 1,$$



Winternitz

Signature generation

Message M and hash function H such that:

$$H(M) = d = \underbrace{b_{t-1}}_w \parallel \cdots \parallel \underbrace{b_{t-t_1}}_w \quad \text{where } 0 \leq \text{int}(b_i) < 2^w - 1,$$

$$c = \sum_{i=t-t_1}^{t-1} 2^w - 1 - b_i,$$



Winternitz

Signature generation

Message M and hash function H such that:

$$H(M) = d = \underbrace{b_{t-1}}_w \parallel \cdots \parallel \underbrace{b_{t-t_1}}_w \quad \text{where } 0 \leq \text{int}(b_i) < 2^w - 1,$$

$$c = \sum_{i=t-t_1}^{t-1} 2^w - 1 - b_i,$$

$$c = \underbrace{b_{t_2-1}}_w \parallel \cdots \parallel \underbrace{b_0}_w,$$



Winternitz

Signature generation

Message M and hash function H such that:

$$H(M) = d = \underbrace{b_{t-1}}_w \parallel \cdots \parallel \underbrace{b_{t_1}}_w \quad \text{where } 0 \leq \text{int}(b_i) < 2^w - 1,$$

$$c = \sum_{i=t-t_1}^{t-1} 2^w - 1 - b_i,$$

$$c = \underbrace{b_{t_2-1}}_w \parallel \cdots \parallel \underbrace{b_0}_w,$$

$$\sigma(M) = (H^{b_{t-1}}(x_{t-1}), \dots, H^{b_0}(x_0)).$$



Winternitz

Signature verification

$$X = (x_{t-1}, \dots, x_1, x_0)$$



Winternitz

Signature verification

$$X = (x_{t-1}, \dots, x_1, x_0)$$

$$\sigma(M) = (H^{b_{t-1}}(x_{t-1}), \dots, H^{b_0}(x_0))$$



Winternitz

Signature verification

$$X = (x_{t-1}, \dots, x_1, x_0)$$

$$\sigma(M) = (H^{b_{t-1}}(x_{t-1}), \dots, H^{b_0}(x_0))$$

$$Y = (H^{2^w-1}(x_{t-1}), \dots, H^{2^w-1}(x_0))$$



Winternitz

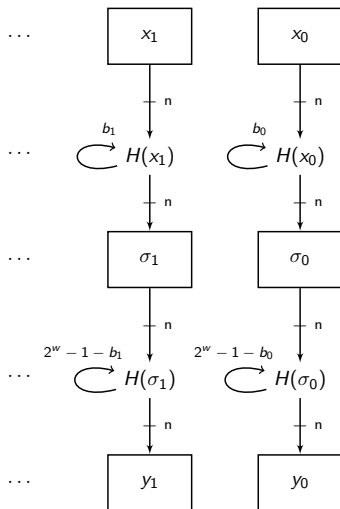
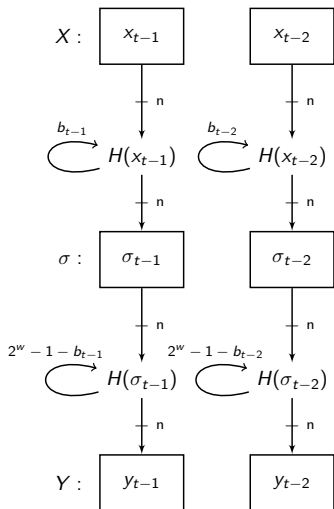
Signature verification

$$X = (x_{t-1}, \dots, x_1, x_0)$$

$$\sigma(M) = (H^{b_{t-1}}(x_{t-1}), \dots, H^{b_0}(x_0))$$

$$Y = (H^{2^w-1}(x_{t-1}), \dots, H^{2^w-1}(x_0))$$

$$\forall y_i \stackrel{?}{=} H^{2^w-1-b_i}(\sigma_i)$$





Winternitz signature scheme

Winternitz signature scheme

Classical Winternitz

Discussion

Related works

Tuning Winternitz

Final remarks



Discussion

Winternitz OTS

Characteristics of the scheme:

- + Post-quantum signature scheme;



Discussion

Winternitz OTS

Characteristics of the scheme:

- + Post-quantum signature scheme;
- + Requires only hash functions;



Discussion

Winternitz OTS

Characteristics of the scheme:

- + Post-quantum signature scheme;
- + Requires only hash functions;
- + Can be used for IoT (hardware acceleration);



Discussion

Winternitz OTS

Characteristics of the scheme:

- + Post-quantum signature scheme;
- + Requires only hash functions;
- + Can be used for IoT (hardware acceleration);
- + **Very fast;**



Discussion

Winternitz OTS

Characteristics of the scheme:

- + Post-quantum signature scheme;
- + Requires only hash functions;
- + Can be used for IoT (hardware acceleration);
- + Very fast;
- Big signatures;



Discussion

Winternitz OTS

Characteristics of the scheme:

- + Post-quantum signature scheme;
- + Requires only hash functions;
- + Can be used for IoT (hardware acceleration);
- + Very fast;
- Big signatures;
- ± Trade-off between signature size and runtime.



Discussion

Motivation

The problem:

- ▶ Hardware can be chosen to accomodate signature requirements;



Discussion

Motivation

The problem:

- ▶ Hardware can be chosen to accomodate signature requirements;
- ▶ Hard to predict resources available for the signature verification;



Discussion

Motivation

The problem:

- ▶ Hardware can be chosen to accomodate signature requirements;
- ▶ Hard to predict resources available for the signature verification;
- ▶ Verification should be easy and fast;



Discussion

Motivation

The problem:

- ▶ Hardware can be chosen to accomodate signature requirements;
- ▶ Hard to predict resources available for the signature verification;
- ▶ Verification should be easy and fast;
- ▶ How can we decrease WOTS signature sizes and keep runtime within an acceptable margin?



Winternitz signature scheme

Winternitz signature scheme

Classical Winternitz

Discussion

Related works

Tuning Winternitz

Final remarks



Related works

Shorter signatures and/or faster signature verification

- ▶ “A one-time signature using run-length encoding” [SV08]



Related works

Shorter signatures and/or faster signature verification

- ▶ “A one-time signature using run-length encoding” [SV08]
 - ▶ Only for small w ;



Related works

Shorter signatures and/or faster signature verification

- ▶ “A one-time signature using run-length encoding” [SV08]
 - ▶ Only for small w ;
 - ▶ A similar non-invasive modification.



Related works

Shorter signatures and/or faster signature verification

- ▶ “A one-time signature using run-length encoding” [SV08]
 - ▶ Only for small w ;
 - ▶ A similar non-invasive modification.
- ▶ “W-OTS⁺ – Shorter Signatures for Hash-Based Signature Schemes” [Hü13]



Related works

Shorter signatures and/or faster signature verification

- ▶ “A one-time signature using run-length encoding” [SV08]
 - ▶ Only for small w ;
 - ▶ A similar non-invasive modification.
- ▶ “W-OTS⁺ – Shorter Signatures for Hash-Based Signature Schemes” [Hü13]
 - ▶ Reduces security requirement of the hash function;



Related works

Shorter signatures and/or faster signature verification

- ▶ “A one-time signature using run-length encoding” [SV08]
 - ▶ Only for small w ;
 - ▶ A similar non-invasive modification.
- ▶ “W-OTS⁺ – Shorter Signatures for Hash-Based Signature Schemes” [Hü13]
 - ▶ Reduces security requirement of the hash function;
 - ▶ Uses more compact hashes.



Related works

Shorter signatures and/or faster signature verification

- ▶ “A one-time signature using run-length encoding” [SV08]
 - ▶ Only for small w ;
 - ▶ A similar non-invasive modification.
- ▶ “W-OTS⁺ – Shorter Signatures for Hash-Based Signature Schemes” [Hü13]
 - ▶ Reduces security requirement of the hash function;
 - ▶ Uses more compact hashes.
- ▶ “Constant-sum fingerprinting for Winternitz one-time signature” [CYK16]



Related works

Shorter signatures and/or faster signature verification

- ▶ “A one-time signature using run-length encoding” [SV08]
 - ▶ Only for small w ;
 - ▶ A similar non-invasive modification.
- ▶ “W-OTS⁺ – Shorter Signatures for Hash-Based Signature Schemes” [Hü13]
 - ▶ Reduces security requirement of the hash function;
 - ▶ Uses more compact hashes.
- ▶ “Constant-sum fingerprinting for Winternitz one-time signature” [CYK16]
 - ▶ Does not require checksum (smaller signature);



Related works

Shorter signatures and/or faster signature verification

- ▶ “A one-time signature using run-length encoding” [SV08]
 - ▶ Only for small w ;
 - ▶ A similar non-invasive modification.
- ▶ “W-OTS⁺ – Shorter Signatures for Hash-Based Signature Schemes” [Hü13]
 - ▶ Reduces security requirement of the hash function;
 - ▶ Uses more compact hashes.
- ▶ “Constant-sum fingerprinting for Winternitz one-time signature” [CYK16]
 - ▶ Does not require checksum (smaller signature);
 - ▶ Map yields faster signature verification;



Related works

Shorter signatures and/or faster signature verification

- ▶ “A one-time signature using run-length encoding” [SV08]
 - ▶ Only for small w ;
 - ▶ A similar non-invasive modification.
- ▶ “W-OTS⁺ – Shorter Signatures for Hash-Based Signature Schemes” [Hü13]
 - ▶ Reduces security requirement of the hash function;
 - ▶ Uses more compact hashes.
- ▶ “Constant-sum fingerprinting for Winternitz one-time signature” [CYK16]
 - ▶ Does not require checksum (smaller signature);
 - ▶ Map yields faster signature verification;
 - ▶ Signature generation scales too fast.



Related works

Our proposal

- ▶ Allow further runtime configuration;



Related works

Our proposal

- ▶ Allow further runtime configuration;
- ▶ Trade-off between signature generation or verification speed;



Related works

Our proposal

- ▶ Allow further runtime configuration;
- ▶ Trade-off between signature generation or verification speed;
- ▶ Non-invasive, no (significant) changes to WOTS.



Tuning Winternitz

Winternitz signature scheme

Tuning Winternitz

Padding

Randomized hashing

Final remarks



WOTS-B

Padding

Recall that

$$c = \sum_{i=t-t_1}^{t-1} (2^w - 1 - b_i),$$



WOTS-B

Padding

Recall that

$$c = \sum_{i=t-t_1}^{t-1} (2^w - 1 - b_i),$$

$$c_{max} = \sum_{i=t-t_1}^{t-1} (2^w - 1 - 0) = t_1(2^w - 1),$$



WOTS-B

Padding

Recall that

$$c = \sum_{i=t-t_1}^{t-1} (2^w - 1 - b_i),$$

$$c_{max} = \sum_{i=t-t_1}^{t-1} (2^w - 1 - 0) = t_1(2^w - 1),$$

$$n_c = \lceil \log_2 c_{max} \rceil = \lceil \log_2 t_1(2^w - 1) \rceil.$$

Unused checksum bits (n_u) for various combinations of w and n .

w	n	n_c	n_u	t_2w	w	n	n_c	n_u	t_2w
2	128	8	2	10	10	128	14	6	20
	192	9	1			192	15	5	
	256	9	1			256	15	5	
	512	10	2			512	16	4	
4	128	9	3	12	12	128	16	8	24
	192	10	2			192	16	8	
	256	10	2			256	17	7	
	512	11	1			512	18	6	
6	128	11	1	12	14	128	18	10	28
	192	11	1			192	18	10	
	256	12	0			256	19	9	
	512	13	5			512	20	8	
8	128	12	4	16	16	128	19	13	32
	192	13	3			192	20	12	
	256	13	3			256	20	12	
	512	14	2			512	21	11	



WOTS-B

Padding

No padding is needed only for the cases where:

- ▶ $w = 3$ and $n = 128$;
- ▶ $w = 5$ and $n = 128$;
- ▶ $w = 6$ and $n = 256$;
- ▶ $w = 7$ and $n = 512$.

WOTS-B

Padding

No padding is needed only for the cases where:

- ▶ $w = 3$ and $n = 128$;
- ▶ $w = 5$ and $n = 128$;
- ▶ $w = 6$ and $n = 256$;
- ▶ $w = 7$ and $n = 512$.

Thus, we propose

$$c^p = c + (2^{n_u} - 1)2^{n_c} = c + 2^{t_2 w} - 2^{n_c}.$$

That is, a padding with 1s.

Tuning Winternitz

Winternitz signature scheme

Tuning Winternitz

Padding

Randomized hashing

Final remarks



WOTS-R

Randomized hashing

Consider the values

$$R \in \mathbb{N}^*,$$

$$\lambda = (\lambda_r : 0 \leq r \leq R),$$

where λ_r can be a nonce or a counter.



WOTS-R

Randomized hashing

- Compute all $d_r = H(M \parallel \lambda_r) = (b_{r,t-1}, \dots, b_{r,t-t_1}) = \mathcal{B}_{r,1}$;



WOTS-R

Randomized hashing

- ▶ Compute all $d_r = H(M \parallel \lambda_r) = (b_{r,t-1}, \dots, b_{r,t-t_1}) = \mathcal{B}_{r,1}$;
- ▶ Compute $S = \{\sum_{b \in \mathcal{B}_{r,1}} : 0 \leq r \leq R\}$;



WOTS-R

Randomized hashing

- ▶ Compute all $d_r = H(M \parallel \lambda_r) = (b_{r,t-1}, \dots, b_{r,t-t_1}) = \mathcal{B}_{r,1}$;
- ▶ Compute $S = \{\sum_{b \in \mathcal{B}_{r,1}} : 0 \leq r \leq R\}$;
- ▶ Choose $\max(S)$ for faster signature verification;



WOTS-R

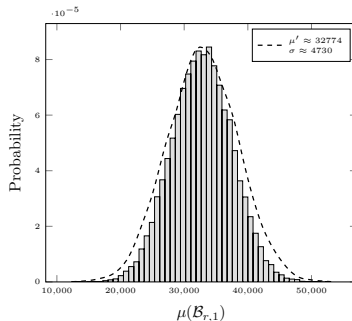
Randomized hashing

- ▶ Compute all $d_r = H(M \parallel \lambda_r) = (b_{r,t-1}, \dots, b_{r,t-t_1}) = \mathcal{B}_{r,1}$;
- ▶ Compute $S = \{\sum_{b \in \mathcal{B}_{r,1}} : 0 \leq r \leq R\}$;
- ▶ Choose $\max(S)$ for faster signature verification;
- ▶ Choose $\min(S)$ for faster signature generation.



WOTS-R

How do we find R ?



Normalized histogram of $\mu(\mathcal{B}_{r,1})$, with $w = 16$, $m = 256$ and $0 \leq r \leq 2^{16}$.



WOTS-R

How do we find R ?

Using a Z-table, we have that

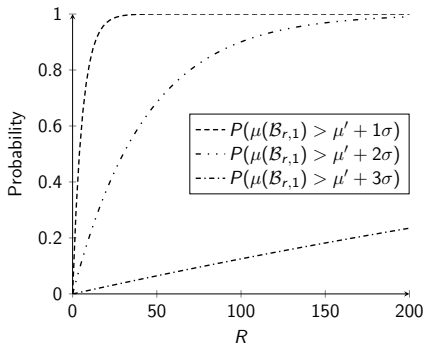
$$P(\mu(\mathcal{B}_{r,1}) > \mu' + 1\sigma) = 0.1587;$$

$$P(\mu(\mathcal{B}_{r,1}) > \mu' + 2\sigma) = 0.0228;$$

$$P(\mu(\mathcal{B}_{r,1}) > \mu' + 3\sigma) = 0.0013.$$

WOTS-R

How do we find R ?



Probability given by the binomial distribution, where we have $\approx 99\%$ of obtaining the desired value if $R \in \{25, 200, 3500\}$.



Final remarks

Experiment results

WOTS	49.65%	50.35%	47.16%	52.84%
WOTS-B	51.04%	48.96%	52.71%	47.29%
WOTS-R	56.72%	43.28%	69.2%	30.8%
WOTS-BR	58.11%	41.89%	74.76%	25.24%

Applications of H
($w = 4, R = 25$)

Applications of H
($w = 16, R = 3500$)

Average distributed cost of H over thousands of signature generation and signature verification steps.



Conclusion

- ▶ Our method is compatible with WOTS+, XMSS and variations;
- ▶ We have tested our method with XMSS, giving very similar results to standalone WOTS;
- ▶ We can use our method for faster signature generation **or** faster signature verification;
- ▶ WOTS-B is a minor change in the checksum padding;
- ▶ WOTS-R is completely non-invasive.

Thank you!



Final remarks

Security remarks

- ▶ WOTS-B: unused bits have no semantics in WOTS security;
- ▶ WOTS-R: no modification to the underlying algorithm;
- ▶ WOTS-R: `shattered.io` uses an attack that can be attempted against this proposal. However, this can be mitigated by using safer hash functions for the message digest.

References



J. P. Cruz, Y. Yatani, and Y. Kaji.

Constant-sum fingerprinting for Winternitz one-time signature.

In *2016 International Symposium on Information Theory and Its Applications (ISITA)*, pages 703–707, October 2016.



A. Hülsing.

W-OTS⁺ – shorter signatures for hash-based signature schemes.

In A. Youssef, A. Nitaj, and A. E. Hassanien, editors, *Progress in Cryptology – AFRICACRYPT 2013*, volume 7918 of *Lecture Notes in Computer Science*, pages 173–188, June 2013.



R. Steinwandt and V. I. Villányi.

A one-time signature using run-length encoding.

Information Processing Letters, 108(4):179–185, October 2008.