

Hash Functions and the Boomerang Attack

ECRYPT Hash Workshop 2007 - Barcelona

Antoine Joux^{1,3} **Thomas Peyrin**^{2,3}

¹ DGA

² France Télécom R&D

³ University of Versailles

May 29, 2007

Outline

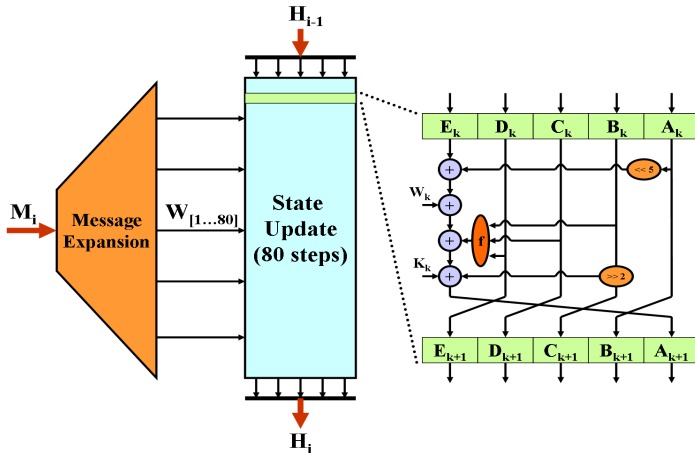
- 1 Introduction
- 2 The (Amplified) Boomerang Attack
- 3 Application to SHA-1
- 4 Conclusion

Outline

- 1 Introduction
- 2 The (Amplified) Boomerang Attack
- 3 Application to SHA-1
- 4 Conclusion

The SHA-1 hash function (1)

Merkle-Damgård + Davies-Meyer mode.



The SHA-1 hash function (2)

Message expansion:

$$W_i = \begin{cases} M_i, & \text{for } 0 \leq i \leq 15 \\ (W_{i-3} \oplus W_{i-8} \oplus W_{i-14} \oplus W_{i-16}) \lll 1, & \text{for } 16 \leq i \leq 79 \end{cases}$$

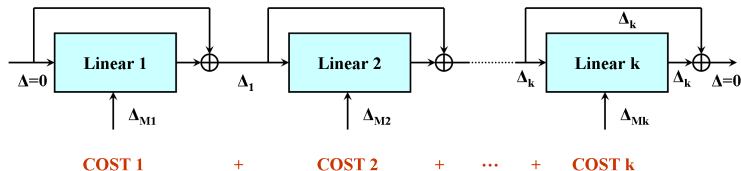
Boolean functions:

step i	$f_i(B, C, D)$
$1 \leq i \leq 20$	$f_{IF} = (B \wedge C) \oplus (\overline{B} \wedge D)$
$21 \leq i \leq 40$	$f_{XOR} = B \oplus C \oplus D$
$41 \leq i \leq 60$	$f_{MAJ} = (B \wedge C) \oplus (B \wedge D) \oplus (C \wedge D)$
$61 \leq i \leq 80$	$f_{XOR} = B \oplus C \oplus D$



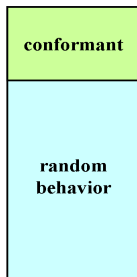
Collision attack against SHA-0 (Biham et al.)

- **local collision**: insert a perturbation and correct it! Then find **perturbation and corrections vectors** such that the overall difference mask satisfies the message expansion.
- **multi-block technique**: you can use several blocks to find a collision.

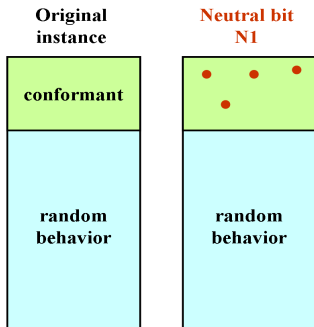


The neutral bits

**Original
instance**

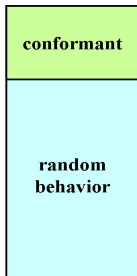


The neutral bits

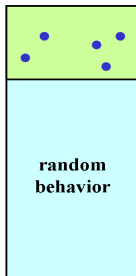


The neutral bits

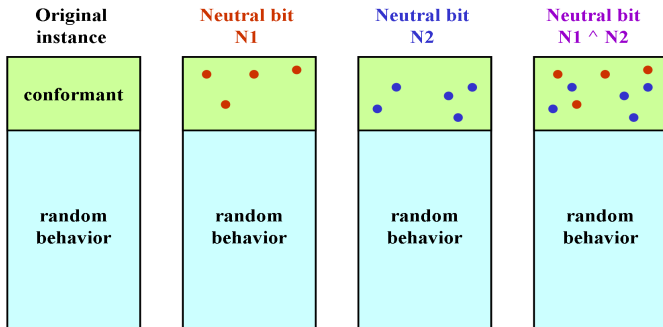
**Original
instance**



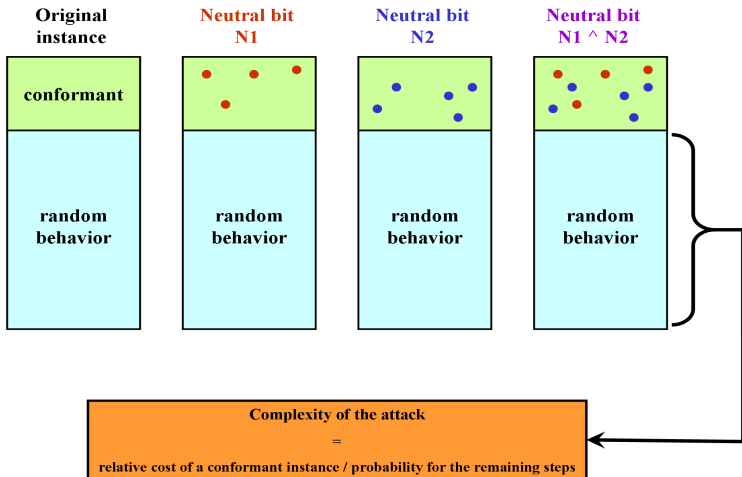
**Neutral bit
N2**



The neutral bits

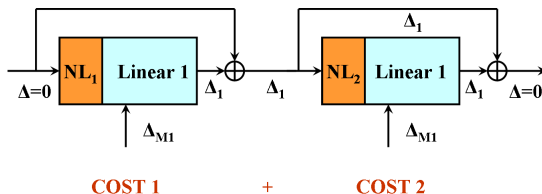


The neutral bits

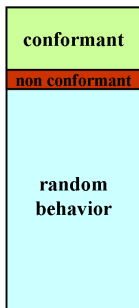


Collision attack against SHA-1 (Wang et al.)

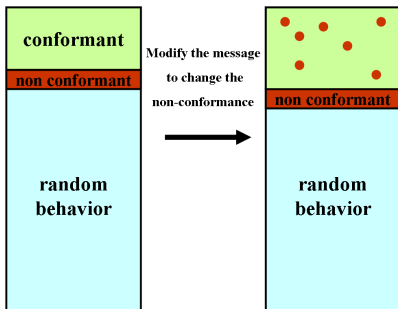
- modify (by hand!) the first steps of the differential path
⇒ **non-linear part**.
- find (by hand!) the **sufficient conditions** such that everything goes as expected
⇒ evaluate the probability of the differential path.
- 2^{69} message modifications (improved to 2^{63} but not published) [Wang, Yin, Yu – 2005].



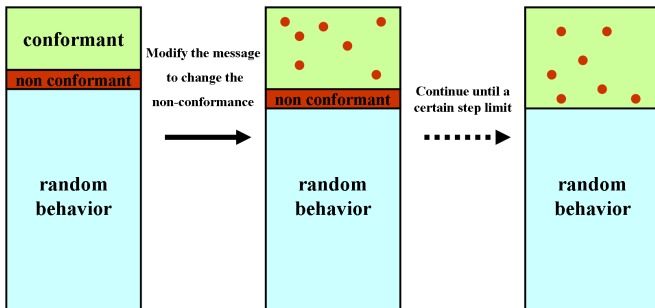
Wang et al.'s attacks: the message modifications



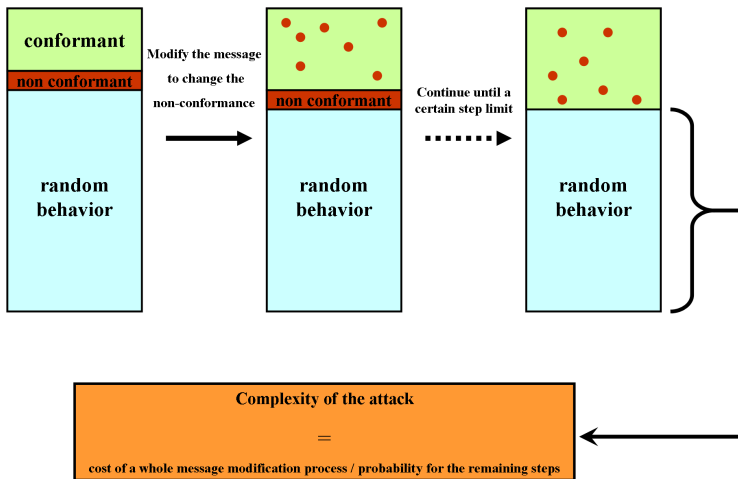
Wang et al.'s attacks: the message modifications



Wang et al.'s attacks: the message modifications



Wang et al.'s attacks: the message modifications



New attacks

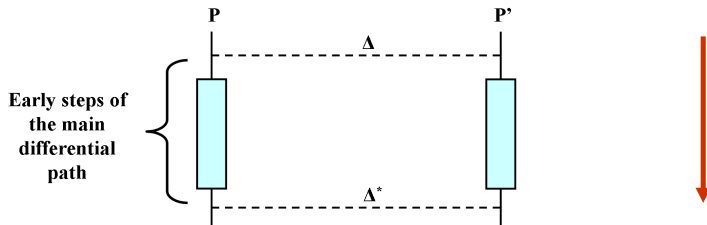
Wang et al. found everything by hand! Can we provide more theoretical explanations of what is happening ?

- a better way of evaluating the **probability of a diff. path** [*De Cannière, Rechberger* – 2006].
- automatic and heuristic **search of non linear parts** [*De Cannière, Rechberger* – 2006].
- finding **sufficient conditions** with Gröbner Basis [*Sugita, Kawazoe, Imai* – 2007].
- finding **message modifications** with Gröbner Basis [*Sugita, Kawazoe, Imai* – 2007].
- a 70-step collision [*De Cannière, Mendel, Rechberger* – 2007].

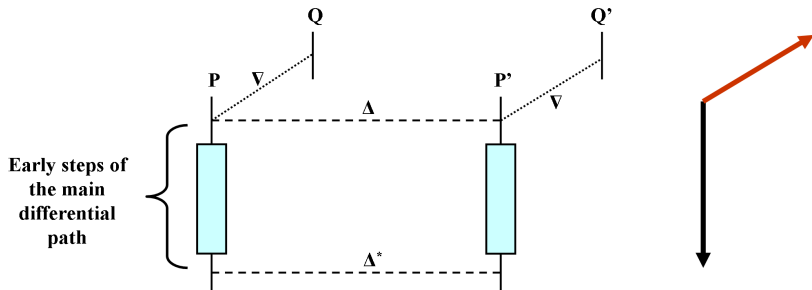
Outline

- 1 Introduction
- 2 The (Amplified) Boomerang Attack**
- 3 Application to SHA-1
- 4 Conclusion

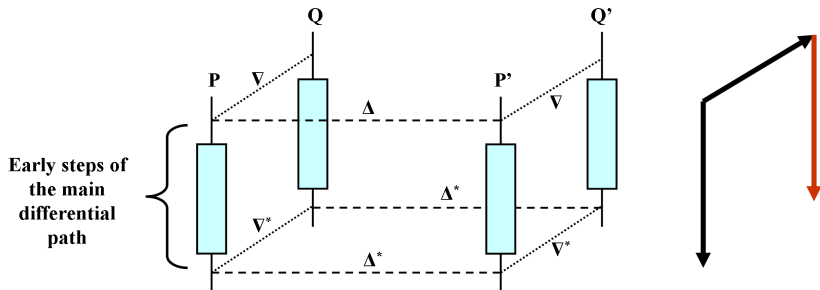
The (amplified) boomerang attack for hash functions (1)



The (amplified) boomerang attack for hash functions (1)



The (amplified) boomerang attack for hash functions (1)



The (amplified) boomerang attack for hash functions (2)

We call the small differential path **auxiliary differential path**.

Two possibilities of use:

- neutral bits approach: instantiate a message pair and check if there is good auxiliary differential paths
⇒ **generalization of neutral bits**.
- explicit conditions approach: **before** instantiating the message pair, fix some bits so that you will be sure that very good auxiliary differential paths exist
⇒ **allows you to find very powerful neutral bits!**

For t auxiliary differential paths, **you get 2^t conformant pairs of messages for free** (with an independence assumption, true in practice).



Outline

- 1 Introduction
- 2 The (Amplified) Boomerang Attack
- 3 Application to SHA-1**
- 4 Conclusion

A useful tool: the local collision

$$A_{i+1} = (A_i \ll 5) + f_i(A_{i-1}, A_{i-2} \gg 2, A_{i-3} \gg 2) + (A_{i-4} \gg 2) + K_i + W_i.$$

step	type	constraints
$i + 1$	no carry	$W_i^l = a, A_{i+1}^l = a$

i	A_i	W_i
-1:	-----	
00:	-----	-----a--
01:	-----a--	-----
02:	-----	-----
03:	-----	-----
04:	-----	-----
05:	-----	-----
06:	-----	-----



A useful tool: the local collision

$$A_{i+1} = (A_i \ll 5) + f_i(A_{i-1}, A_{i-2} \gg 2, A_{i-3} \gg 2) + (A_{i-4} \gg 2) + K_i + W_i.$$

step	type	constraints
$i + 1$	no carry	$W_i^j = a, A_{i+1}^j = a$
$i + 2$	correction	$W_{i+1}^{j+5} = \bar{a}$

i	A_i	W_i
-1:	-----	
00:	-----	-----a--
01:	-----a--	----- \bar{a} -----
02:	-----	-----
03:	-----	-----
04:	-----	-----
05:	-----	-----
06:	-----	-----



A useful tool: the local collision

$$A_{i+1} = (A_i \lll 5) + f_i(A_{i-1}, A_{i-2} \ggg 2, A_{i-3} \ggg 2) + (A_{i-4} \ggg 2) + K_i + W_i.$$

step	type	constraints
$i + 1$	no carry	$W_i^j = a, A_{i+1}^j = a$
$i + 2$	correction	$W_{i+1}^{j+5} = \bar{a}$
$i + 3$	no correction	$A_{i-1}^{j+2} = A_i^{j+2}$
	correction	$A_{i-1}^{j+2} \neq A_i^{j+2}, W_{i+2}^j = \bar{a}$

i	A_i	W_i
-1:	-----d-----	
00:	-----d-----	-----a---
01:	-----a-----	----- \bar{a} -----
02:	-----	-----
03:	-----	-----
04:	-----	-----
05:	-----	-----
06:	-----	-----



A useful tool: the local collision

$$A_{i+1} = (A_i \lll 5) + f_i(A_{i-1}, A_{i-2} \ggg 2, A_{i-3} \ggg 2) + (A_{i-4} \ggg 2) + K_i + W_i.$$

step	type	constraints
$i + 1$	no carry	$W_i^j = a, A_{i+1}^j = a$
$i + 2$	correction	$W_{i+1}^{j+5} = \bar{a}$
$i + 3$	no correction	$A_{i-1}^{j+2} = A_i^{j+2}$
$i + 4$	no correction	$A_{i+2}^{j-2} = 0$
	correction	$A_{i+2}^{j-2} = 1, W_{i+3}^{j-2} = \bar{a}$

i	A_i	W_i
-1:	-----d----	
00:	-----d----	-----a--
01:	-----a--	----- \bar{a} ----
02:	-----1	-----
03:	-----	----- \bar{a}
04:	-----	-----
05:	-----	-----
06:	-----	-----



A useful tool: the local collision

$$A_{i+1} = (A_i \lll 5) + f_i(A_{i-1}, A_{i-2} \ggg 2, A_{i-3} \ggg 2) + (A_{i-4} \ggg 2) + K_i + W_i.$$

step	type	constraints
$i + 1$	no carry	$W_i^j = a, A_{i+1}^j = a$
$i + 2$	correction	$W_{i+1}^{j+5} = \bar{a}$
$i + 3$	no correction	$A_{i-1}^{j+2} = A_i^{j+2}$
$i + 4$	correction	$A_{i+2}^{j-2} = 1, W_{i+3}^{j-2} = \bar{a}$
$i + 5$	no correction	$A_{i+3}^{j-2} = 1$
	correction	$A_{i+3}^{j-2} = 0, W_{i+4}^{j-2} = \bar{a}$

i	A_i	W_i
-1:	-----d-----	
00:	-----d-----	-----a--
01:	-----a--	----- \bar{a} -----
02:	-----1	----- \bar{a} -----
03:	-----0	----- \bar{a} -----
04:	-----	----- \bar{a} -----
05:	-----	-----
06:	-----	-----

A useful tool: the local collision

$$A_{i+1} = (A_i \ll 5) + f_i(A_{i-1}, A_{i-2} \gg 2, A_{i-3} \gg 2) + (A_{i-4} \gg 2) + K_i + W_i.$$

step	type	constraints
$i + 1$	no carry	$W_i^j = a, A_{i+1}^j = a$
$i + 2$	correction	$W_{i+1}^{j+5} = \bar{a}$
$i + 3$	no correction	$A_{i-1}^{j+2} = A_i^{j+2}$
$i + 4$	correction	$A_{i+2}^{j-2} = 1, W_{i+3}^{j-2} = \bar{a}$
$i + 5$	correction	$A_{i+3}^{j-2} = 0, W_{i+4}^{j-2} = \bar{a}$
$i + 6$	correction	$W_{i+5}^{j-2} = \bar{a}$

i	A_i	W_i
-1:	-----d----	
00:	-----d----	-----a--
01:	-----a--	----- \bar{a} ----
02:	-----1	-----
03:	-----0	----- \bar{a}
04:	-----	----- \bar{a}
05:	-----	----- \bar{a}
06:	-----	----- \bar{a}



Building auxiliary differential paths

	W_0 to W_{15}	W_{16} to W_{31}
perturbation mask	1010000000100000	
differences on W^j	1010000000100000	0000000010110110
differences on W^{j+5}	0101000000010000	0000000001011011
differences on W^{j-2}	0001111100000011	0000000000001110

i	A_i	W_i
-1:	-----d---	
00:	-----d---	-----a--
01:	-----e-a--	-----ā--
02:	-----e-1	-----b--
03:	-----b-0	-----b̄--ā
04:	-----0	-----ā
05:	-----0	-----ā
06:	-----	-----b̄
07:	-----	-----b̄
08:	-----	-----
09:	-----f---	-----
10:	-----f---	-----c--
11:	-----c--	-----c̄--
12:	-----0	-----
13:	-----0	-----
14:	-----	-----ā
15:	-----	-----c̄



Building auxiliary differential paths

	W_0 to W_{15}	W_{16} to W_{31}
perturbation mask	1010000000100000	
differences on W^j	1010000000100000	0000000010110110
differences on W^{j+5}	010100000010000	000000001011011
differences on W^{j-2}	0001111100000011	0000000000001110

i	A_i	W_i
-1:	-----d---	
00:	-----d---	-----a--
01:	-----e-a--	-----ā--
02:	-----e-1	-----b--
03:	-----b-0	-----b̄--ā
04:	-----0	-----ā
05:	-----0	-----ā
06:		-----b̄
07:		-----b̄
08:		-----
09:	-----f---	-----
10:	-----f---	-----c--
11:	-----c---	-----c̄--
12:	-----0	-----
13:	-----0	-----
14:	-----	-----ā
15:	-----	-----c̄



Building auxiliary differential paths

	W_0 to W_{15}	W_{16} to W_{31}
perturbation mask	1010000000 1 00000	
differences on W^j	1010000000 1 00000	0000000010110110
differences on W^{j+5}	0101000000 0 10000	0000000001011011
differences on W^{j-2}	00011111000000 1 1	0000000000001110

i	A_i	W_i
-1:	-----d---	
00:	-----d---	-----a--
01:	-----e-a--	----- \bar{a} ---
02:	-----e--1	-----b--
03:	-----b-0	----- \bar{b} --- \bar{a}
04:	-----0	----- \bar{a} ---
05:	-----0	----- \bar{a} ---
06:	-----	-----b--
07:	-----	----- \bar{b} ---
08:	-----	-----
09:	-----f---	-----
10:	-----f---	-----c--
11:	-----c--	----- \bar{c} ---
12:	-----0	-----
13:	-----0	-----
14:	-----	----- \bar{c} ---
15:	-----	----- \bar{c} ---



Building auxiliary differential paths

	W_0 to W_{15}	W_{16} to W_{31}
perturbation mask	1010000000100000	
differences on W^j	1010000000100000	0000000010110110
differences on W^{j+5}	0101000000010000	0000000001011011
differences on W^{j-2}	0001111100000011	0000000000001110

i	A_i	W_i
-1:	-----d---	
00:	-----d---	-----a--
01:	-----e-a--	-----ā--
02:	-----e--1	-----b--
03:	-----b-0	-----b̄--ā
04:	-----0	-----ā
05:	-----0	-----ā
06:	-----	-----b̄
07:	-----	-----b̄
08:	-----	-----
09:	-----f---	-----
10:	-----f---	-----c--
11:	-----c--	-----c̄--
12:	-----0	-----
13:	-----0	-----
14:	-----	-----c̄
15:	-----	-----c̄



Placing auxiliary differential paths

i	A_i	W_i
-4:	00101001010011011100100101000111	
-3:	00000111100001000110010101100010	
-2:	11011000010000101001111101011111	
-1:	01011011110111101101101111010001	
00:	01000010101101110111101110011011	1uu11101100111110110--0111111011
01:	n1n010111001011001001-0100100110	nuu101-10001011--11111101u1n0n1
02:	1nu11--0111101111101101111111u1	--n11-----0-10-1111000110n0111uu
03:	nnu00----0-00-0110000110111110n	x-nn-1--1--01010001001--1u111001
04:	u010u11-0--0010010110-1010un0u1	uu-u0-----11-0--1011001n1n10nu
05:	1001u00-0--00000000001u00011010	nn-u0-----11010111--1--11n100u1
06:	011unnnnnnnnnnnnnnn1--110n001uu	00n-----1-1--1--00111100011001
07:	u110-01000000u010110nu111u01010n	1nu001-----1--1-100-1-10-un-0n-
08:	1111010111111--011unu110-0--nu1	--un0-----11-----u0111nu
09:	-0010--1--1--01-0u-10nnnnu01010	--u0-----1--1001-u1--100
10:	-----1--1--0--01-101nu1111u10	xxu00-----0--1--1--0--1--u----n-
11:	0-----0--1--1--0n-100nn0u1n0	-xn--1--0-0--1-0---11-0010--x-
12:	0--0-----0-0-0--01-010n1-nn	x-----11-----u
13:	00-----0-0--0--00100n0n-00	--10-----11-----0--1n1---
14:	-0--0-----10001u0un-	--1-----1--0-0--1--000--xn
15:	n-----unnn1101	-x-10-----1--0-0--1--0u-n--u-
16:	--1-----1--nu001	-n0-----11-----1u0----
17:	n-0-----111-0n	xxn-----1--1u-x--n-
18:	-11-----101-	x-u1-----0-----0--0--
19:	-----u-	x-----11n-----
20:	-----	--x-----x



Results

- we can use boomerang attacks with neutral bits or message modifications if we carefully check that the auxiliary paths remain valid.
- message modifications can be costly and the 2^{63} attack is not yet published.
- works well with neutral bits.
- we expect an improvement of a factor 32 (5 auxiliary paths) on the known attacks against SHA-1.

If you are interested in the details, see our paper!

Outline

- 1 Introduction
- 2 The (Amplified) Boomerang Attack
- 3 Application to SHA-1
- 4 Conclusion**

Yet another way of using freedom degrees ...

- boomerang attack for hash functions is nothing more than another way of cleverly using the freedom degrees from the message.
- message modifications, neutral bits, Klima's tunnels for MD5, auxiliary differentials are closely related.
- they all have pros and cons:

	message modifications	neutral bits	auxiliary paths
speed cost	big	medium	small
freedom degrees cost	medium	small	big
range	medium	small	long



... but freedom degrees are not unlimited!

- we can not use all those techniques independently!
- **twofold waste of freedom degrees**: or we use a lot of freedom degrees for a small gain, or some freedom degrees are left unused.
- it would be great to find a way to use **exactly** what we need from all those techniques.
- not trivial since we need to settle the long range characteristics first, which imposes a lot (too much ?) of constraints.
- maybe a generalization of those techniques may achieve this ?

Thank you!