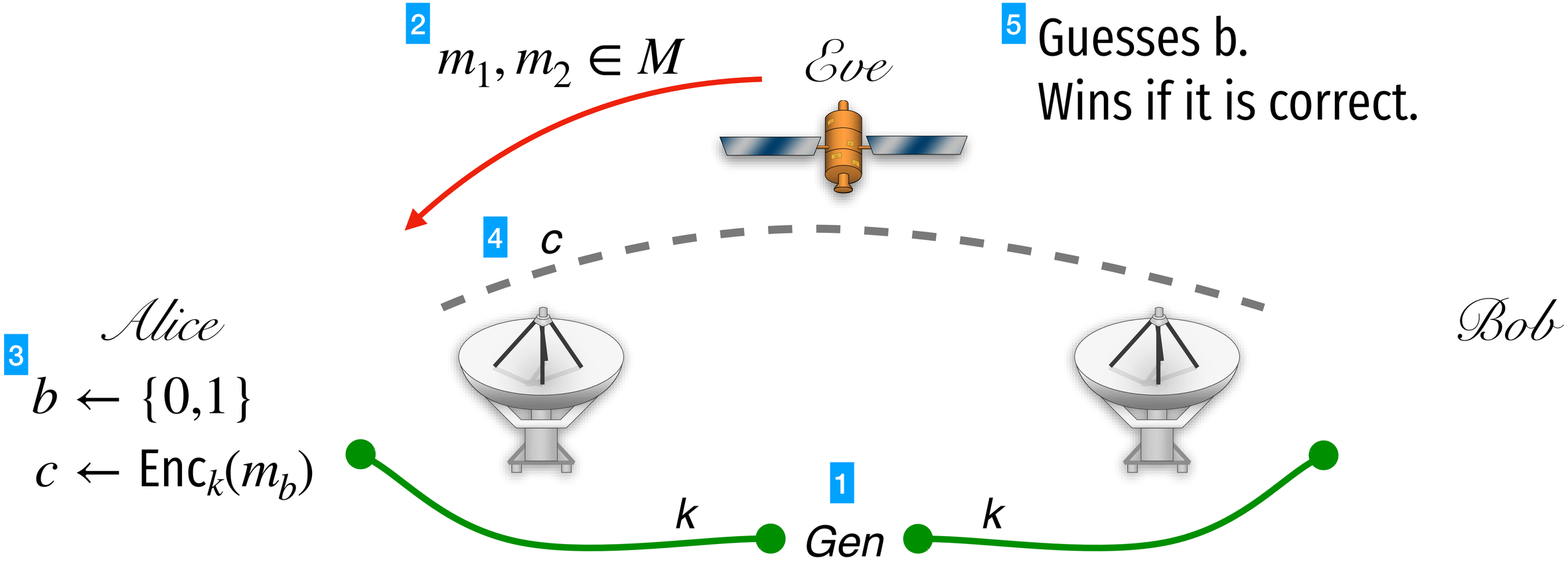
2550 Intro to cybersecurity

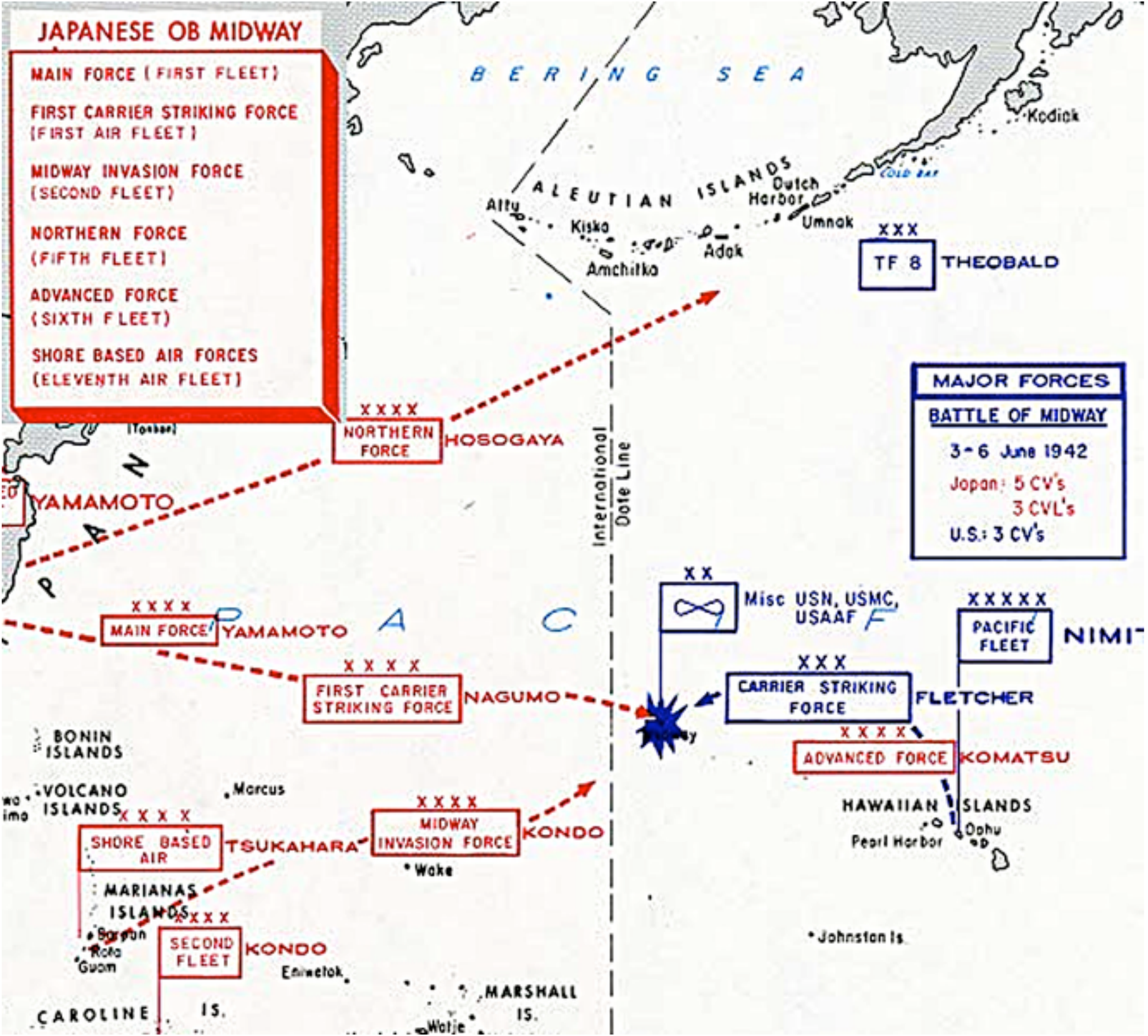
Public key Crypto

abhi shelat

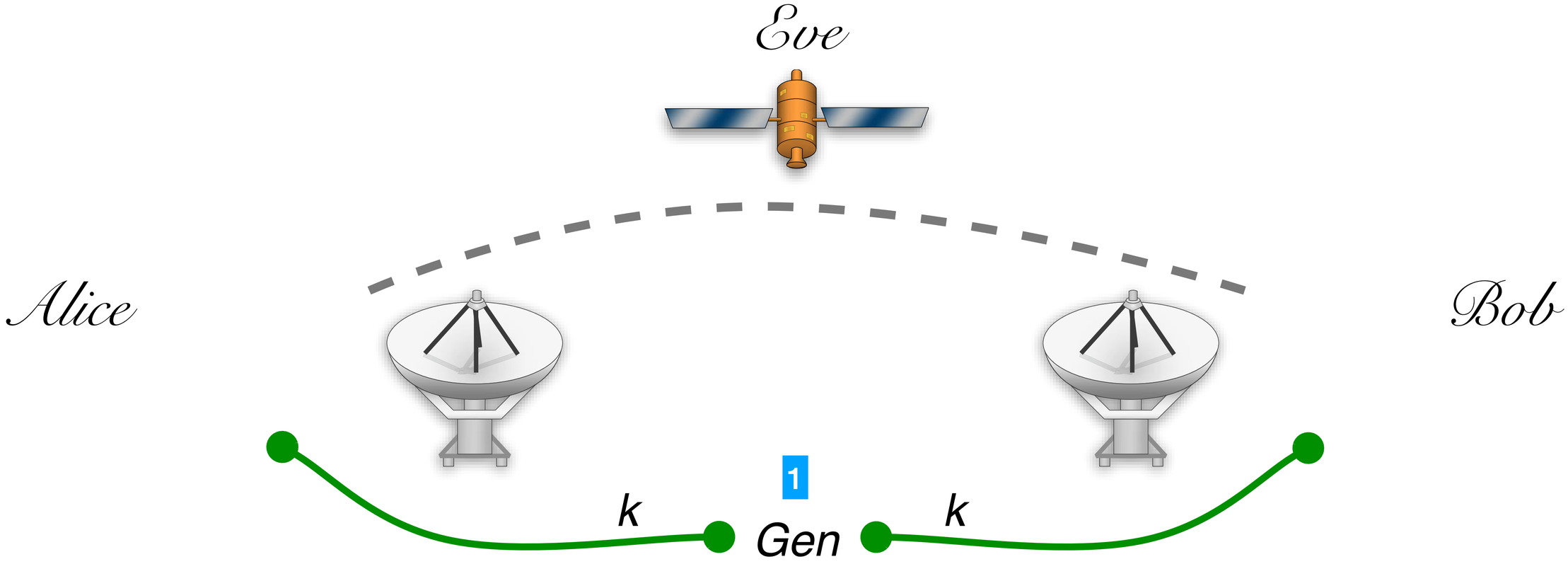
Recap

Is this game strong enough to capture all feasible attacks?

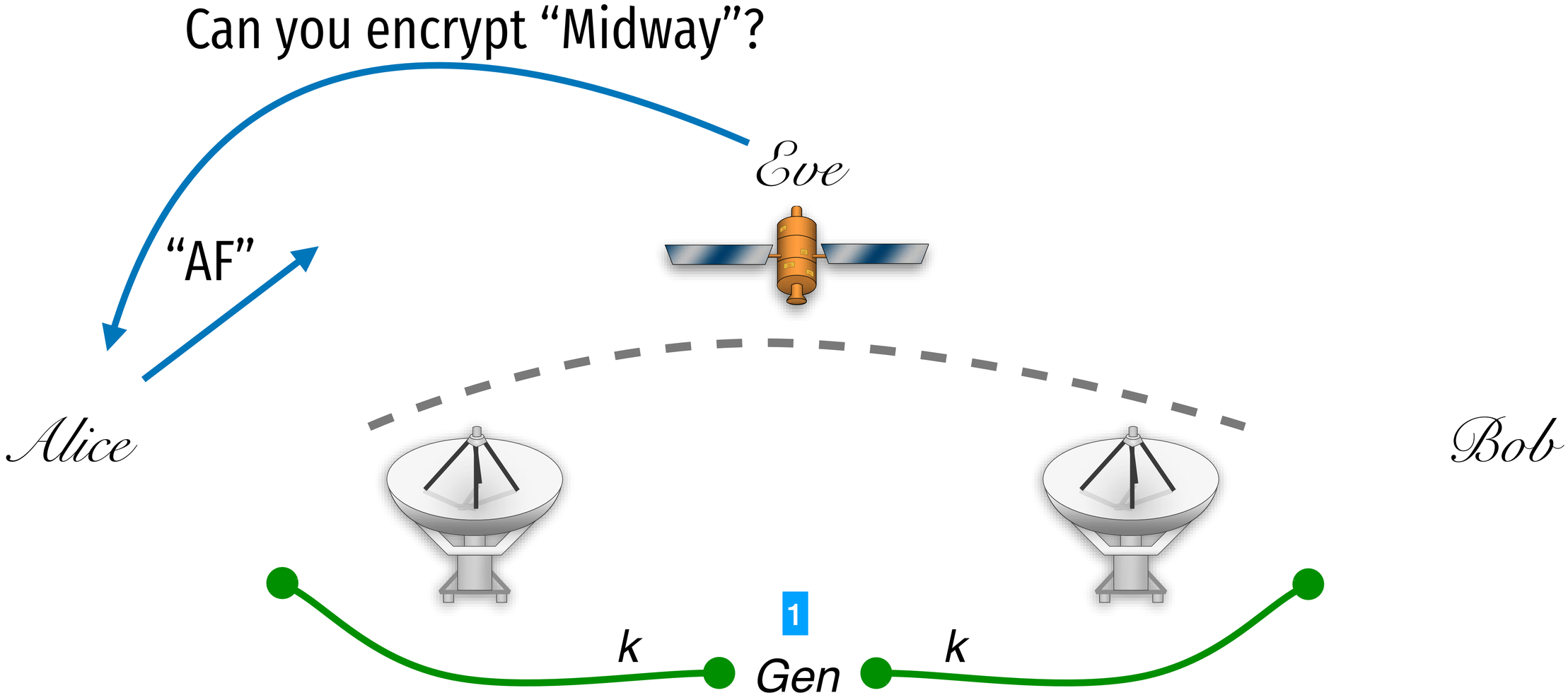



JAPANESE OB MIDWAY
MAIN FORCE (FIRST FLEET)
FIRST CARRIER STRIKING FORCE (FIRST AIR FLEET)
MIDWAY INVASION FORCE (SECOND FLEET)
NORTHERN FORCE (FIFTH FLEET)
ADVANCED FORCE (SIXTH FLEET)
SHORE BASED AIR FORCES (ELEVENTH AIR FLEET)
BERING SEA
Kodiak
ALEUTIAN ISLANDS
Dutch Harbor
Umnak
Attu
Kiska
Amchitka
Adak
XXX
TF 8
THEOBALD
MAJOR FORCES
BATTLE OF MIDWAY
3 - 6 June 1942
Japan: 5 CV's
3 CVL's
U.S.: 3 CV's
XXXX
NORTHERN FORCE
HOSOGAYA
YAMAMOTO
International Date Line
XX
Misc USN, USMC, USAAF
XXXXX
PACIFIC FLEET
NIMI
XXXX
MAIN FORCE
YAMAMOTO
XXXX
FIRST CARRIER STRIKING FORCE
NAGUMO
XXX
CARRIER STRIKING FORCE
FLETCHER
XXXX
ADVANCED FORCE
KOMATSU
BONIN ISLANDS
VOLCANO ISLANDS
Marcus
XXXX
SHORE BASED AIR
TSUKAHARA
XXXX
MIDWAY INVASION FORCE
KONDO
Wake
HAWAIIAN ISLANDS
Oahu
Pearl Harbor
MARIANAS ISLANDS
Saipan
Rota
Guam
XXX
SECOND FLEET
KONDO
Eniwetok
Johnston Is
CAROLINE IS.
MARSHALL IS.
Wotje

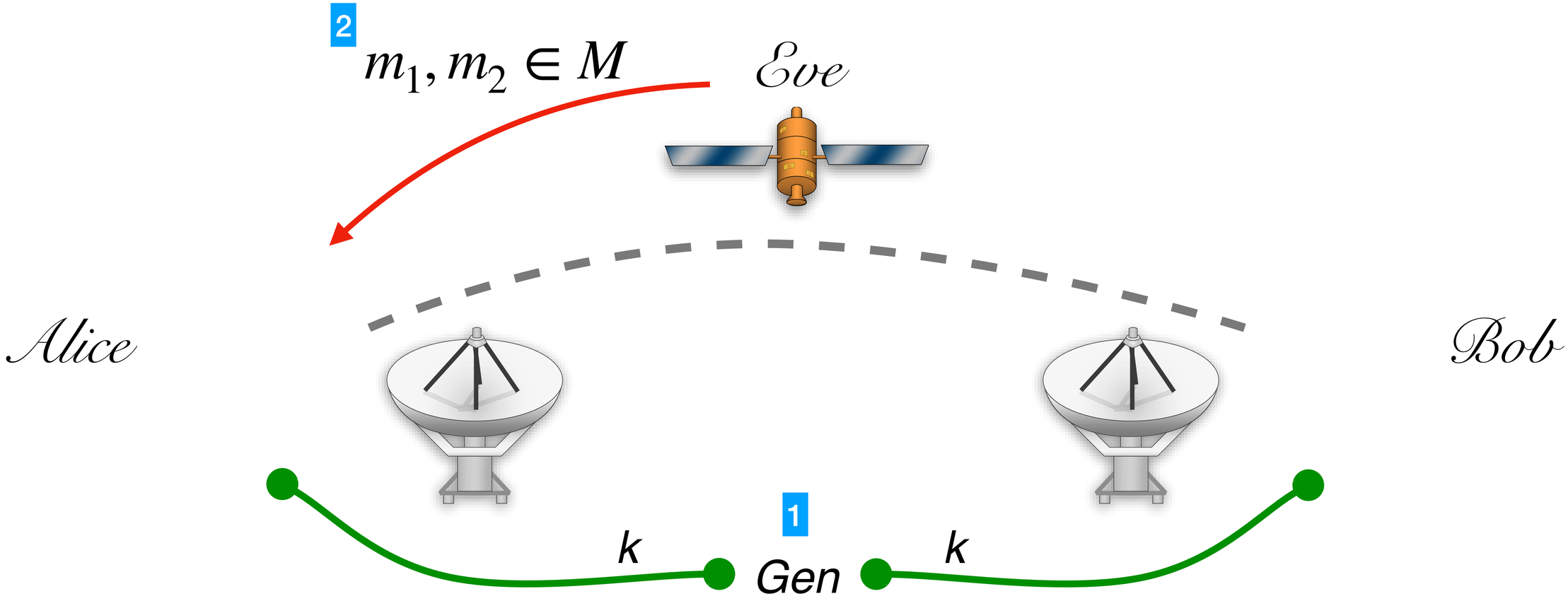
A summary of the attack



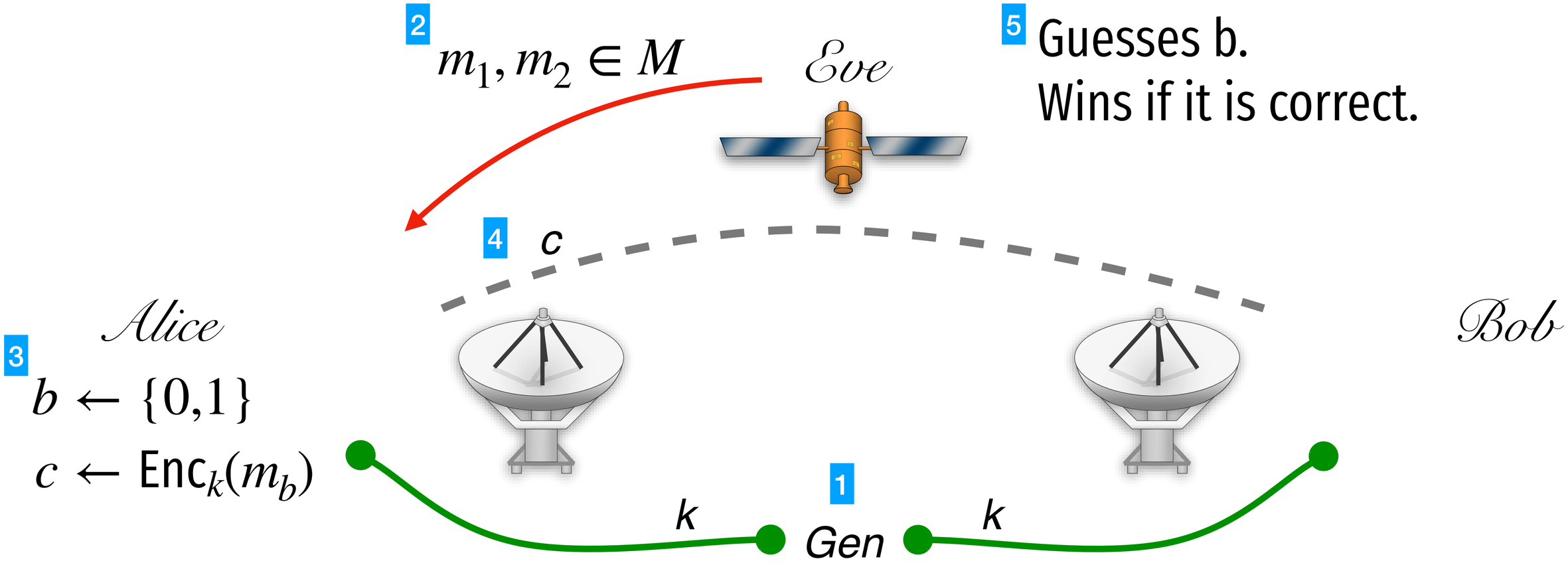
A summary of the attack



A summary of the attack

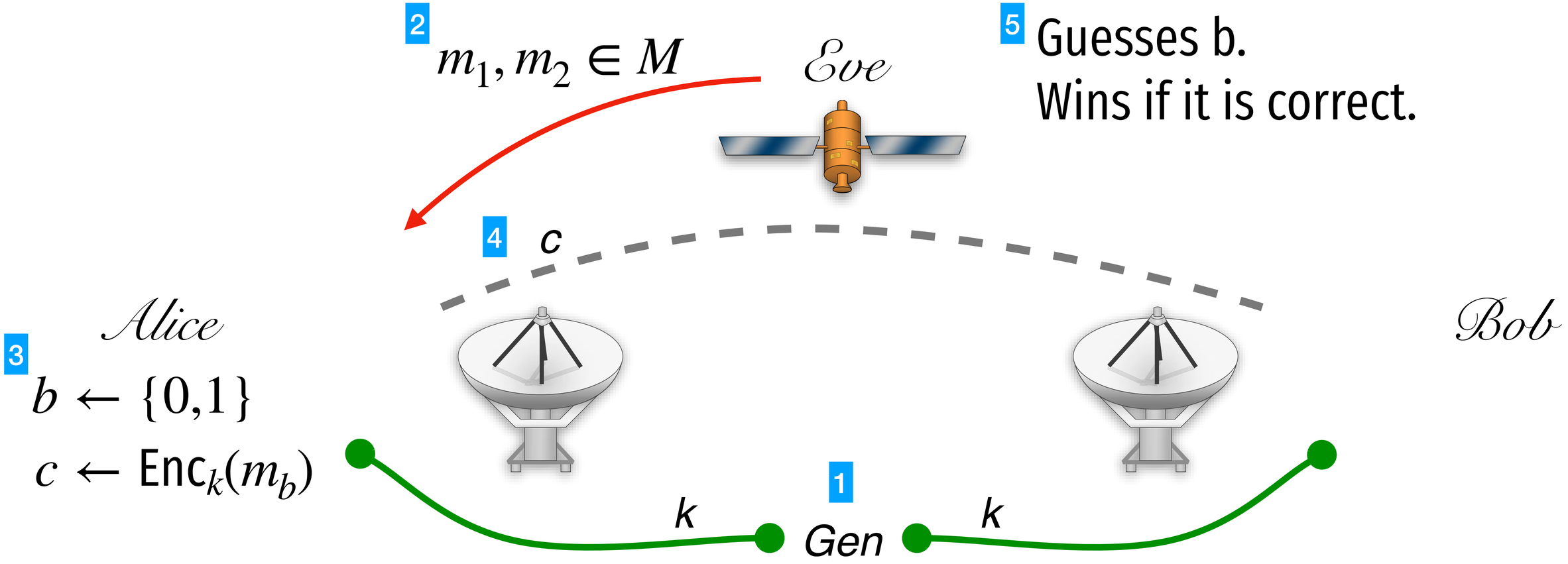


A summary of the attack



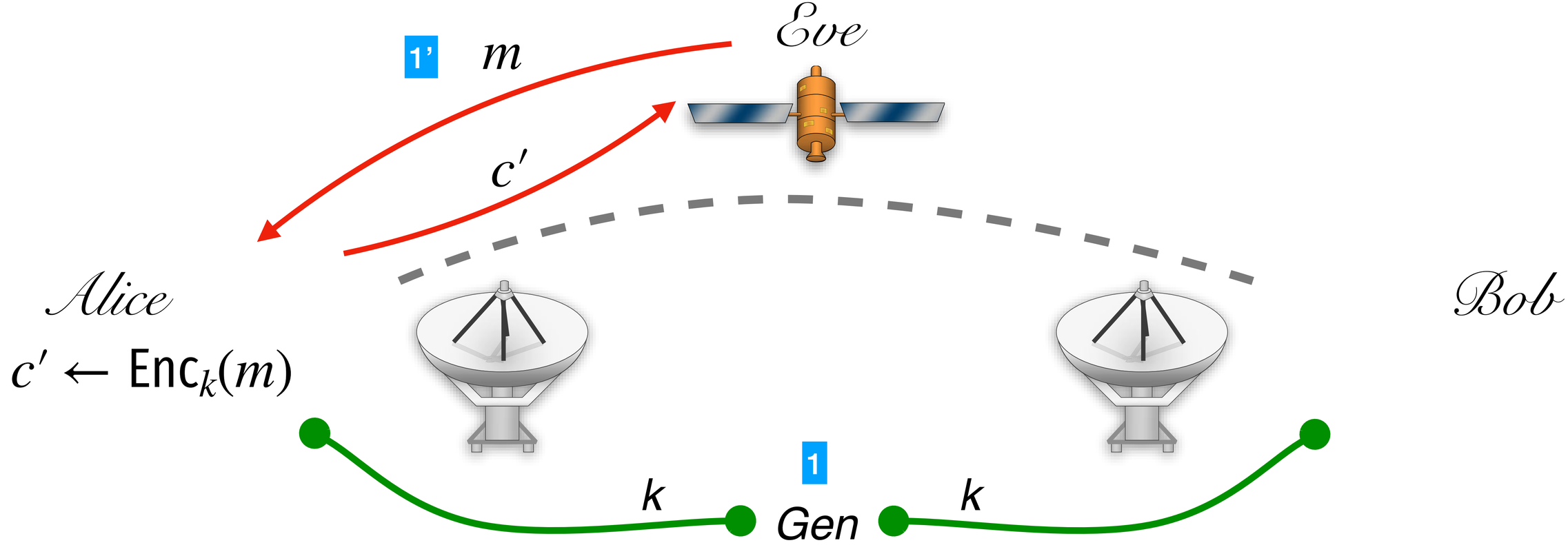
A summary of the attack

Eve was allowed to ask for encryptions of arbitrary messages before step2.

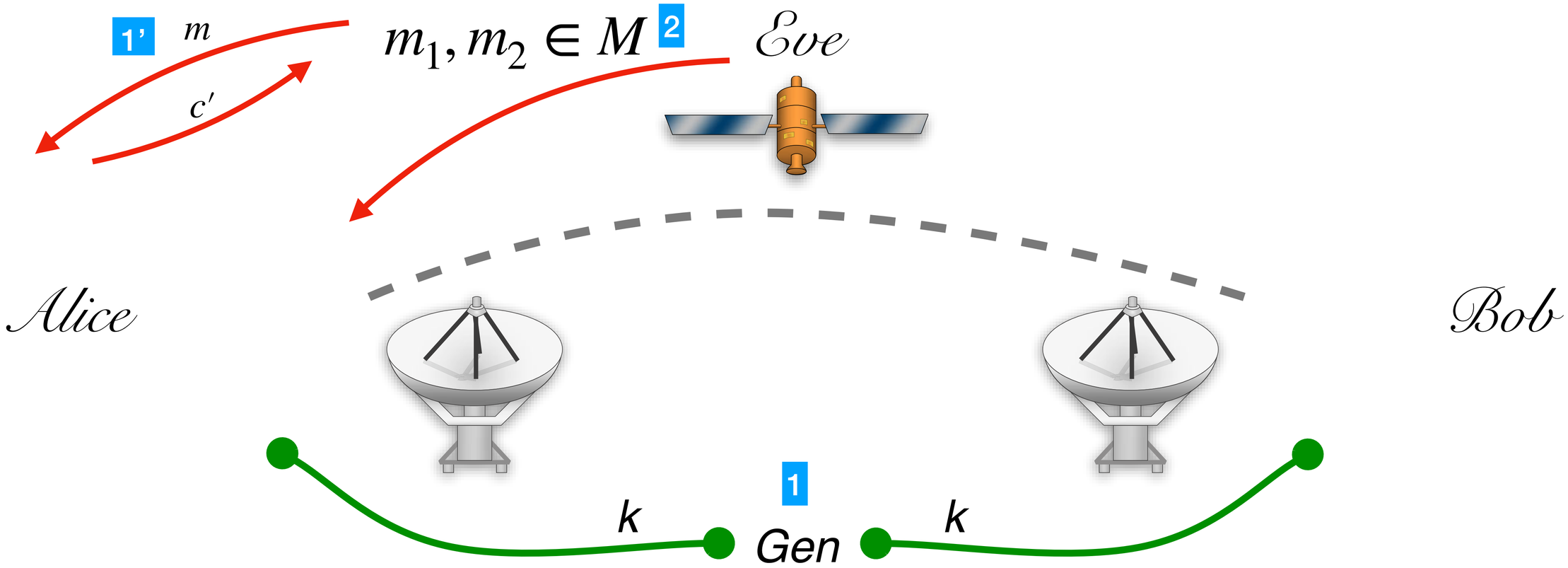


IND-CPA attack for Symmetric Enc

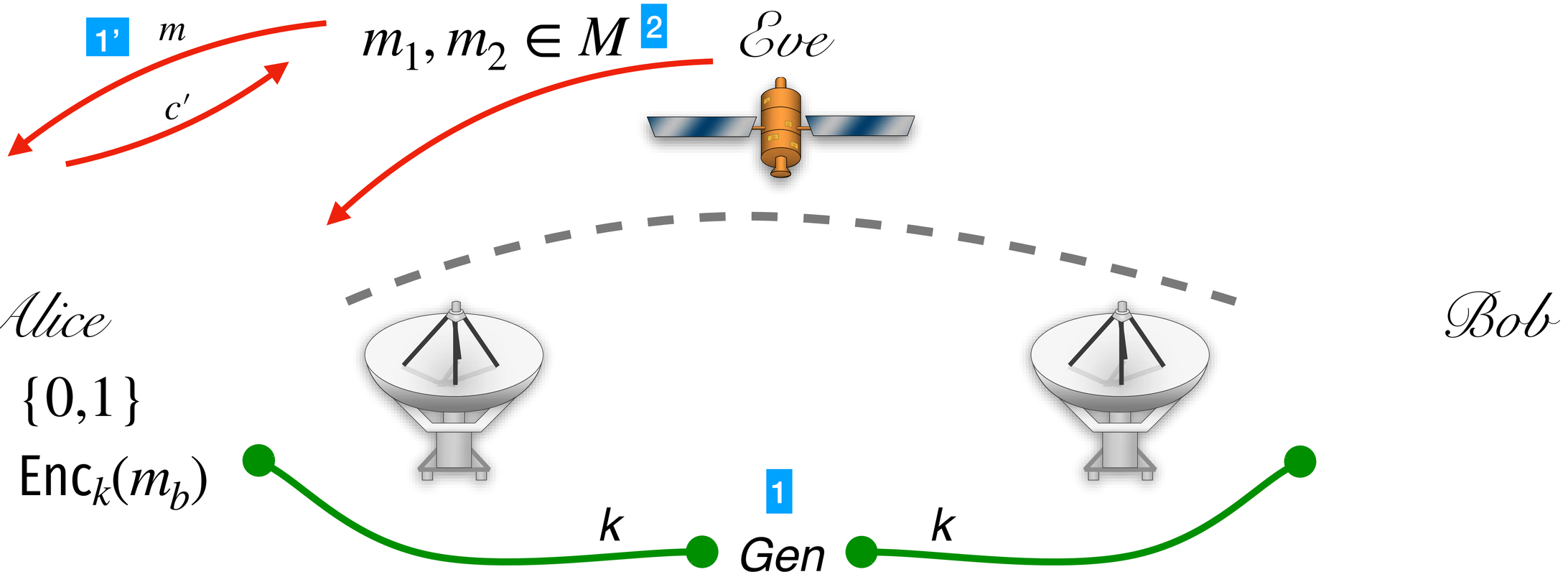
IND-CPA attack for Symmetric Enc



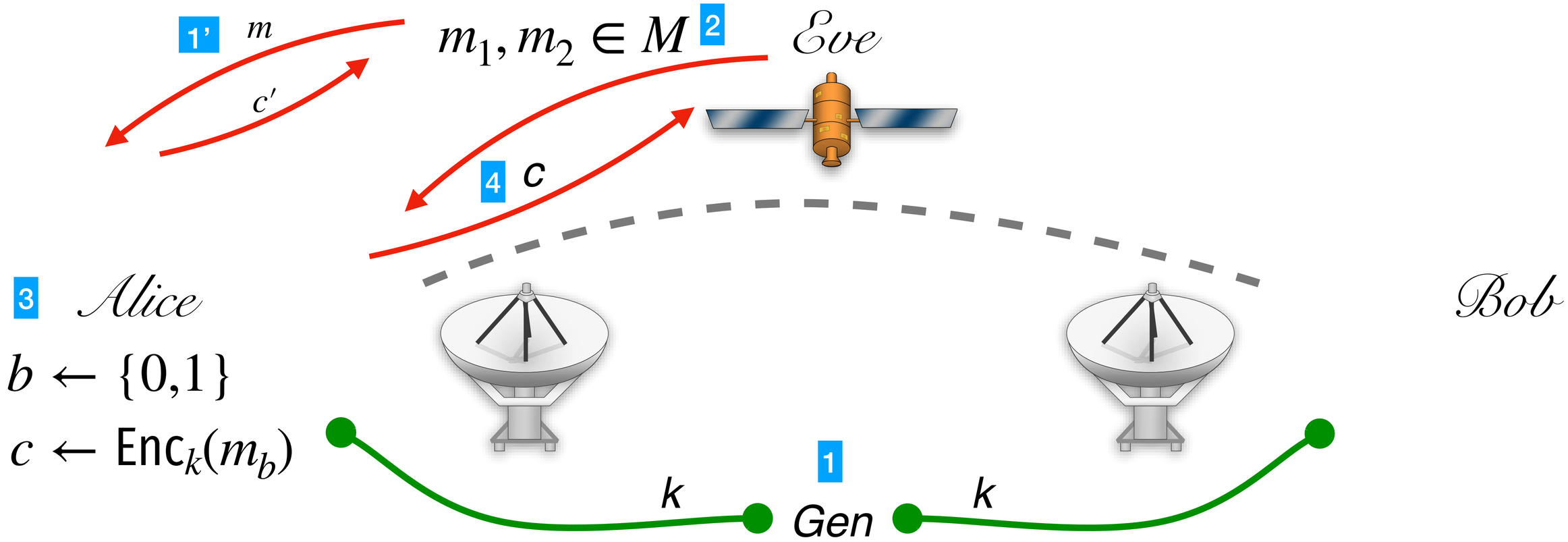
IND-CPA attack for Symmetric Enc



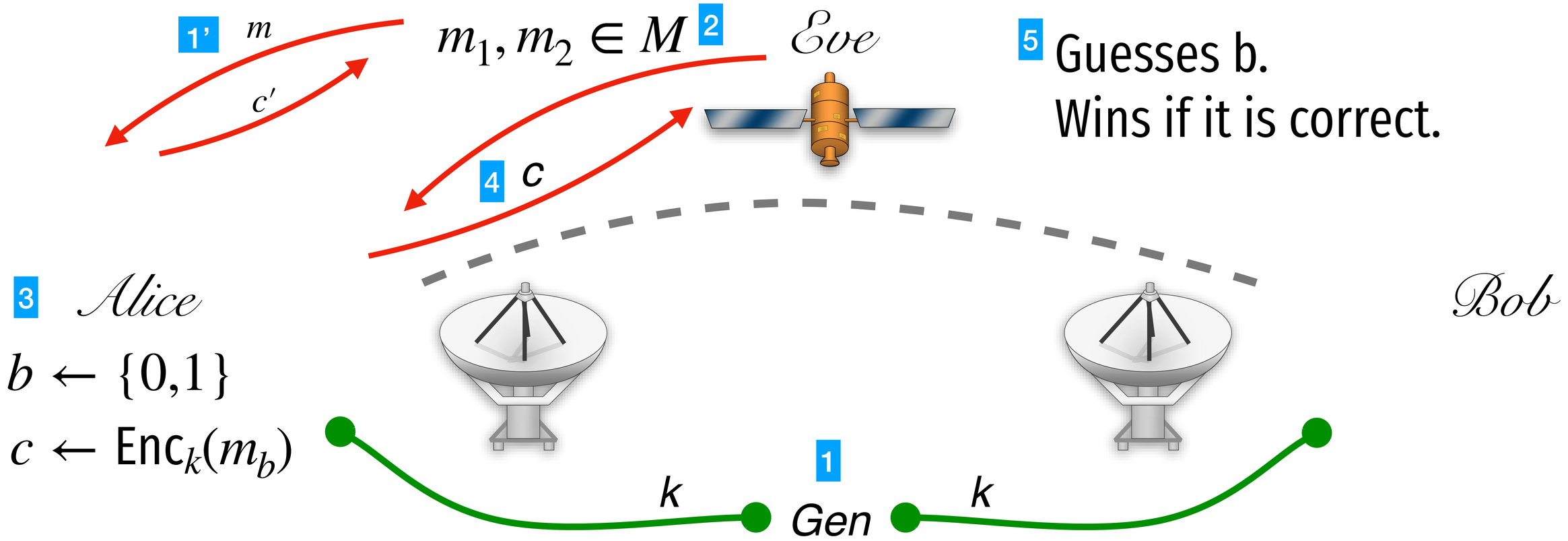
IND-CPA attack for Symmetric Enc



IND-CPA attack for Symmetric Enc



IND-CPA attack for Symmetric Enc



To satisfy IND-CPA, Enc must be randomized.

Enc(k, “hello”)

Enc(k, “hello”)

To satisfy IND-CPA, Enc must be randomized.

Enc(k, “hello”)

Enc(k, “hello”)

If the encryption of a message is always the same cipher text,
then the scheme CANNOT be IND-CPA secure!

Theorem: If One-way functions exist, Then IND-CPA secure symmetric encryption exists.

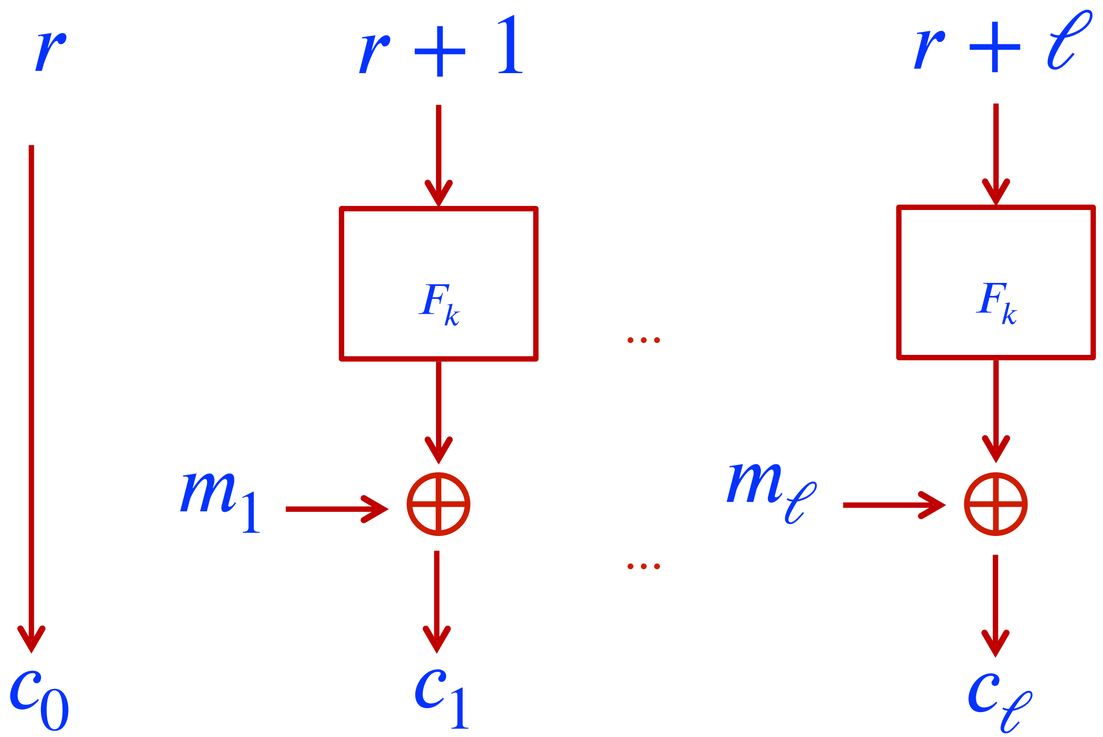
Example of IND-CPA: AES-CTR

Example of IND-CPA: AES-CTR

$$\mathrm{Enc}_k\left(m_1 \cdots m_\ell; r\right)$$
$$= \left(r, F_k(r+1) \oplus m_1, F_k(r+2) \oplus m_2, \ldots, F_k\left(r+\ell\right) \oplus m_\ell\right)$$

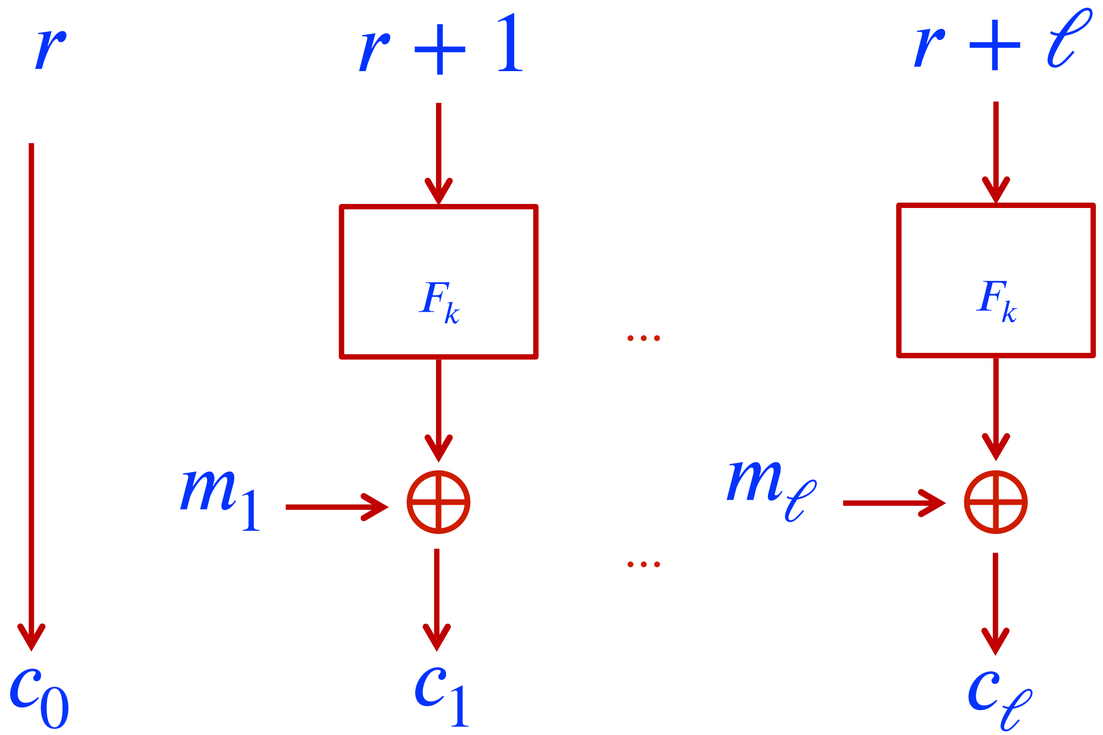
Example of IND-CPA: AES-CTR

$$\mathrm{Enc}_k(m_1 \cdots m_\ell; r)$$
$$= \left(r, F_k(r+1) \oplus m_1, F_k(r+2) \oplus m_2, \ldots, F_k(r+\ell) \oplus m_\ell\right)$$



Example of IND-CPA: AES-CTR

$$\mathrm{Enc}_k(m_1 \cdots m_\ell; r)$$
$$= \left(r, F_k(r+1) \oplus m_1, F_k(r+2) \oplus m_2, \ldots, F_k(r+\ell) \oplus m_\ell\right)$$



Ciphertext expansion is just one block

How to use AES-CTR with openssl

```
$ openssl enc -aes-128-ctr -a
```

Two ways to build PRGS + Symmetric Enc

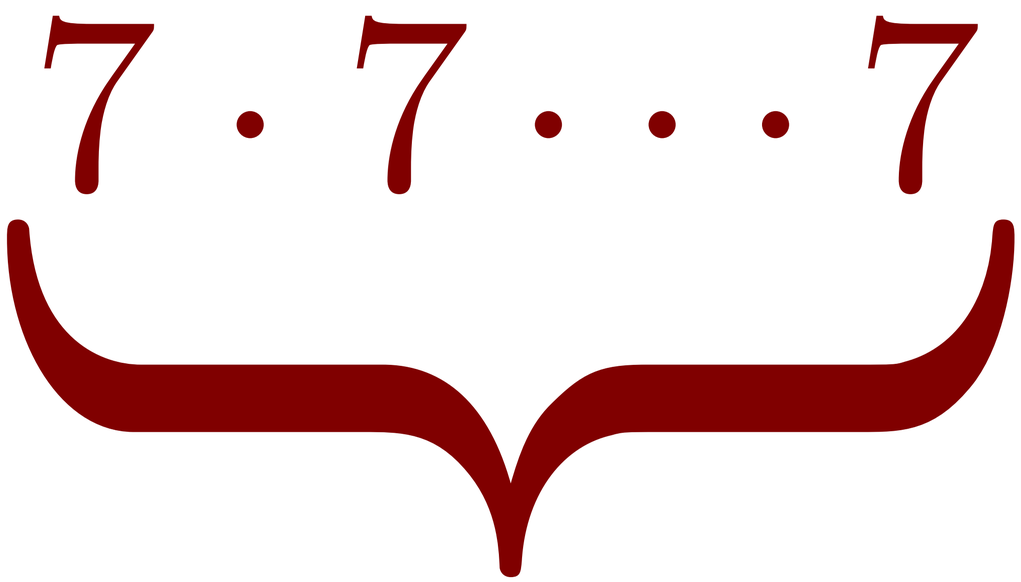
Principled

Heuristic

How can we build pseudo-random generators from principled assumptions?

7

7 * 7



$$7^x$$

7^{1000}

1253256639965718318107554832382734206164985075080986171463495007520970596317381164324488390543515207631986159195515940766858289
8946726302276179083827085457983001511124666120398462435892983257161571801470409630566809750761327366302322689525054138592715842
6088684494082416768617708189592286936039922311125683719215046689156738352590137241554510185855964549927575493247391132548534378
4979788060849510858742020118363623157274201095547829887915300882897118445505002304856384131899471321422439473341992593007356224
9293741945365006149030210512792031443040163685567754913633748132181134967842707609143734504539933734861126116805592935540299282
3192491190360027036112283180935872775214517464013178274657100736321564606838252739601156414628445543663144696050650160812621814
3270626661951727017802002866450238230831859280613713103008292840711412077312806000001

7^{10000}

95587288567444097027721621933777274477156508800837670232809747736882802688984737969351872846915678176303391547032418309440558367061778809896056537787
80399681208530465536899923872083728356691350372430337696619926639213646136205780780799564658253882535717364116449482061428736198739029989324013720616
22798495221711362842179827842727414295003904969814698887647607489396731788861520661760706501004653662221061702731812881848142381806779161139684997815
33857229583655468142483727845972893261563186914448396106549374910394408496075970728927209606377135009929343770889544886395210196985248511900291396723
34581096971834261761008159149534484118177736597236191071022784445587105969769066773943151852497146078670876701272357547133220831824869952253405695472
24637119563739118489076548828690529780180285781790200709105216646950073585170470697772749289033207716010189419921643140458588853251039197496070544889
71652559170240935385129592437066664987443935314310318787709572181947687110460396215414209965523374474058355202693094219280954613682387054722403565561
68312930388055119358800784517897136666191879774303837940606358829794421599264636917853403624967743494135245600686835446913717210276536426354058103443
91477855285718310897324368748177703731383290606646196241766322273949789427186494573916220387132050170812874260685993980785991365062484904774284092738
45322293948595250418133360588306541097125781628428197968967708785261936745801584142161171842336306061944607580316035087158545049258220657354157171176
42951641517044516126295875685241125426213156539867646429188235196051228173319647034882976905185935972581124925543404862657129745082863682768217744513
28996893631934434841791210613977737646500169926470060725939192900341372098057933793890328524106945011721530158679597549366961544323520396108216845856
60293442216332255507058075007157057934008407572852685095108138875038783467527808394272833031297114865113153007797888201808406197212237152401786723674
96637042272860292245006757733008463731247340730605827559431458687544786940933264980621447456906707991547128831033370293831198789730566529222267250464
94001644377943743452783533866658332825699115526102481808742271316354083188217216735072401255075831638502315154196349830971547344213153150365567385039
52579851113060316168285064441232537103590182621079377832843019727603226096866723988722891001182616180230977331759604555319089344641568797218346060486
91762387934893989942083292678611416543611768850763424813437908028375226699829575967165231253786105772454128891004817031720210914641548363847541833079
22685257765508608568562975782300697197204032540698594430286977219892762377295441285181042879127802999024719080613210271422102554740752710289946806735
37259392939342492022814681786038589505372793746046900629086230630259659924116133599305462808406201633629428637101656936527502479985898768332184004335
96639923643807793537485122586379078832995205363197648627390047750400245522036935578273297419752227419104359783605213749942580361012317260930722582974
40735657687656653417040524326319232166510843949286941183532589457466149344002398542045031768982820134480807533788305522366049120476884093493951048644
07345150328046921038566652090310866567690216457225948387309395598104621188922299520026018146333402349443994282604600758932366539153393090504007103991
18863480174135729126354272327526198931847020456672879532506696333611778066210820579248916858197102905558561073173654738527663802764255064566025275401
17971724098601448357463139833373628747458351815403543341999906781590500386735380120235572414794413580542733270524897365599418650411306114136782193558
93476567393224303909082190781300190833131836924007165565422712251679628183801808697438471248270896656566801392106045976601094176105501078995413630627
05900288745023359281714958359096826397307447904564333619769407450980366639836081278482723694195682833281497420595461768778209208730309972518514909989
34287793587314012691376625709193571373268588370102514439184276312405197068399204828411850218556413588259632958205256693710041664679094714984459699883
10316737367710893103311537634234857638996253794816827403174591340923560355044325962134721789323355449602766761015911412461491615405576309511983254641
61885467428416604670527897561905627890990303514190544175156737750586986075245694893976422852835685633907883822672296964630492690390472027126075506211
74274970352102351832642427284726879688473896768703167609747279566439557849880039271927498105906302920460627895414166402998481514856889679663643426632
59247454948386361302515198028514970927235946532045204096515486837699846692599719887881781588315175577292901153095490566264727752914630463195487187579
81842193591589605589369634076907345944834305812025808883916881139977347406610415831245496265129664629242501811263830170572918809855757043839746107727
51107669799661841774127735327529629987980135495274197751725763357367533317639259538639631812862500262581861604130322932796068961074378772900122619610
95075078523249740107441649074205707503339857610479405501354002538172669003503829435619181066724693520857662065505547119775673157892582974345012851224
03345550857364422106089400098097980710060066914254172300542199332763840179799715395210417981973291905376852621563340366608473882951965177128199303553
97583590688781450433887737655547586394649717798423766234759515768872575211201196790591177321395066955978185826696587614884398699723936600276791352446
68433614093303498345026183151219733115820351724151015788117079632416971637874342694800810920986588090282101606161109083273544569683892703076884385369
76695229421549667227488553970025901787267380707303012994912546555721617521680092662101142028491284877797981483652170709353747182682111023619809685273
30949883894783925818255430850346483797946697754305745787159147586676699909367391555177886905439435498431811636138145965247047741495066940582775786312
52301110003288825754780655518167212766883031211722120463766886341101457050139738519752913754699466098465402702788083305676643164175653164229195638479
75365991386486427417012635420756335784238406846840625370460253690959084071991825552481499175959058377594905342179223414288460633222644816783912057037
07075354557112511918073873731274143941499679321050883552677572347377797105743135909768100623080044190543340518440840016792409365173907051366700767393
24486401863133453523692216106044837426872832723526396728877036669789154090875467323512305194341580241450118168534722412097353945900003375105188790288
35718109216677544706167532779894334539341149484795033723281521701291803170133581206157279751790372163062541903141120263356260078563881028828158044322
56679034524428893261896452335957448388704292410919276051488657556172485434602528126371477338338731272695247871214853134420858804219624769529530849648
24305461165483344052069705969562416904959592823688304857706697333346755997888475028602837035242143266422559083501581674445876349033308556314401236564
30541941576610331943099851040885295757540282534806827228478629743216744688778020961699159280663882846234689310077081736602436416683820654352128761662
53949114415001240168660248112047077862925289611603281589332853085867183258825475535891673425352359526353307011017102021845622963926668929672265589093
26151385289475138108952342727756434000169914627664682562543427367805737384621907606506084956620139938655220574272932080283397508049767236316047819413
05576510837522255187398122932575970203218521158723210423508984903210022531681471604161527283425018665371038789874350824128553133183897478964095512132
10937623745552392107065182174128788308985231038174181837246685831115344213967160913112271567741324804620469079393138672209754183205957678245976901221
45557246628826166559252644534293178263695795467108877595954276881268091161226599431968684120556631276490437760589000224109022110256760502207506663688
18628742451124407441779463590967479146039988804969207182365365846712087804226663296973642180785685584316325681616712576674198500848437376727048109601
05239453347719567200067141961335872126244003706089402421142339549257662158399796716927732769103582352343826936693477382089998797977999286844063504765
61759956702127474423874275294409773813374583525458136888505635396193029768685508418714596477252230715588593206795794876020845605080470098683651949698
87758274490615795726597857727249417216377578462655704874289400476685655266879736209501456715267417789381173273996244782286259433553292793669397502330
94318083024928950873135211011911245462183122413017132231172152100517301924058849195801312549213512806000001



7^100000000000000000

Examples Random

Input:

$7^{100\,000\,000\,000\,000\,000}$

Power of 10 representation:

$10^{10^{10^{1.2285776105298 23}}}$

Number length:

84 509 804 001 425 684 decimal digits

$\approx 8.45098 \times 10^{16}$ digits

Last few digits:

...0000000001

Computed by **Wolfram** ***Mathematica***

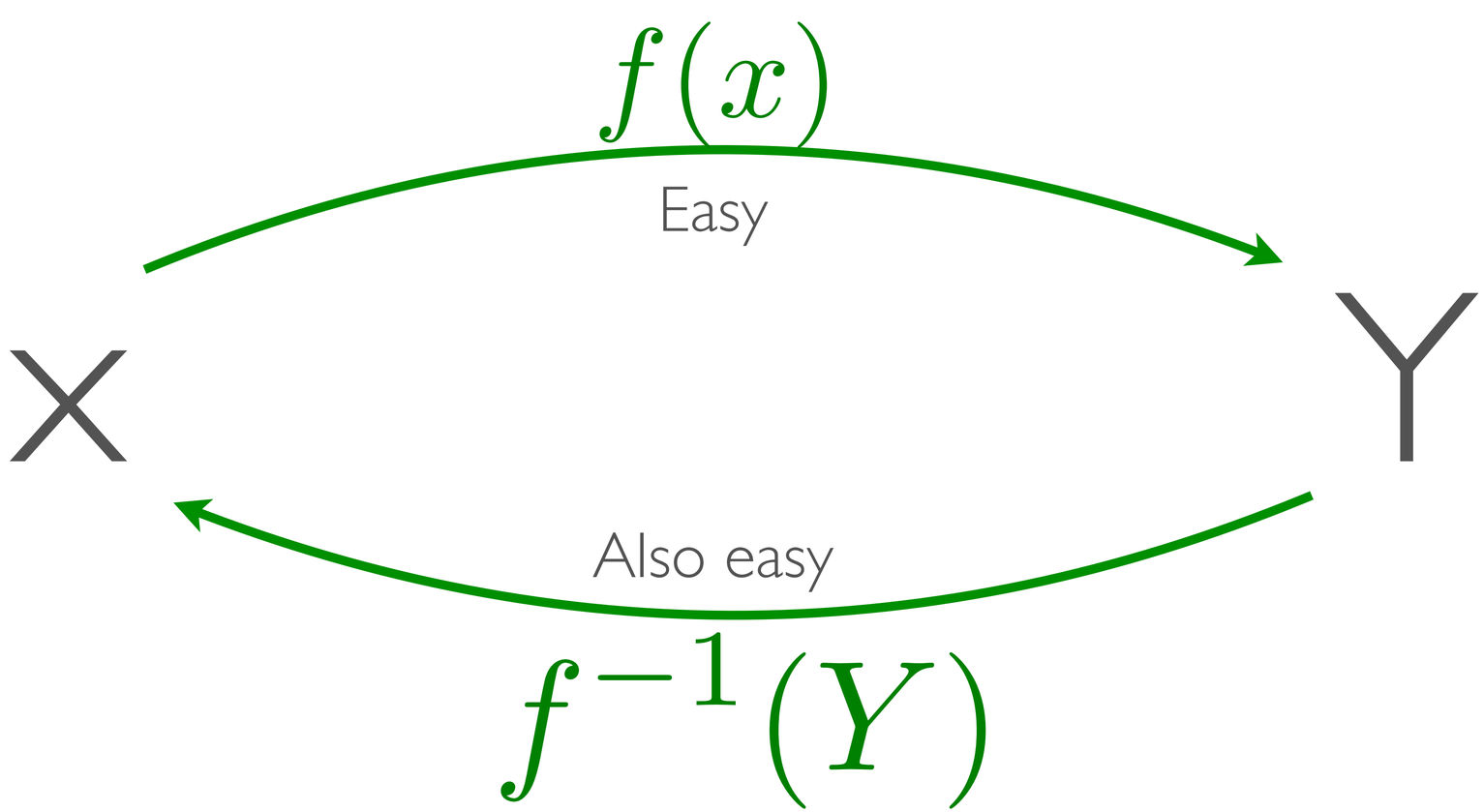
Download page

Inverse problem is EASY:

Given: 1231928

Find X such that $7^X = 1231928$

7.20695710776661638632370146569432966292556545698087919670 51



$$7^x$$

Modular Arithmetic

Multiplication table mod 13:

×	0	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8	9	10	11	12
2	0	2	4	6	8	10	12	1	3	5	7	9	11
3	0	3	6	9	12	2	5	8	11	1	4	7	10
4	0	4	8	12	3	7	11	2	6	10	1	5	9
5	0	5	10	2	7	12	4	9	1	6	11	3	8
6	0	6	12	5	11	4	10	3	9	2	8	1	7
7	0	7	1	8	2	9	3	10	4	11	5	12	6
8	0	8	3	11	6	1	9	4	12	7	2	10	5
9	0	9	5	1	10	6	2	11	7	3	12	8	4
10	0	10	7	4	1	11	8	5	2	12	9	6	3
11	0	11	9	7	5	3	1	12	10	8	6	4	2
12	0	12	11	10	9	8	7	6	5	4	3	2	1

$$\underbrace{7 \cdot 7 \cdots 7}_{x \text{ times}}$$

$$7^x \mod p$$

$$7^2 = 7 \cdot 7 \mod 131$$

$$7^2 = 7 \cdot 7 \mod 131$$

$$= 49$$

$$7^3 = 49 \cdot 7 \mod 131$$

$$343 \mod 131$$

$$7^3 = 49 \cdot 7 \mod 131$$

$$\begin{array}{r|l} & 2 \\ \hline 131 & 343 \\ & 262 \\ \hline & 81 \end{array} \mod 131$$

$$7^3 = 49 \cdot 7 \mod 131$$

$$\begin{array}{r} 2 \\ 131 \overline{)\,343} \\ 262 \\ \hline 81 \end{array} \mod 131$$

$$= 81 \mod 131$$

MOD-EXP

$$(a, x, n) \rightarrow a^x \bmod n$$

MOD-EXP

$$(a, x, n) \rightarrow a^x \bmod n$$

$$a^x \bmod n = \prod_{i=0}^{\ell} x_i a^{2^i} \bmod n$$

MOD-EXP

$$(a, x, n) \to a^x \bmod n$$

Algorithm 2: ModularExponentiation(a, x, n)

Input: $a, x \in [1, n]$

```
1 r ← 1
2 while x > 0 do
3     if x is odd then
4         r ← r · a mod n
5     x ← ⌊x/2⌋
6     a ← a^2 mod n
7 Return r
```

MOD-EXP

$$(a, x, n) \to a^x \bmod n$$

$$a^x \bmod n = \prod_{i=0}^{\ell} x_i a^{2^i} \bmod n$$

Algorithm 2: ModularExponentiation(a, x, n)

Input: $a, x \in [1, n]$

1 $r \leftarrow 1$
2 **while** $x > 0$ **do**
3 **if** x *is odd* **then**
4 $r \leftarrow r \cdot a \bmod n$
5 $x \leftarrow \lfloor x/2 \rfloor$
6 $a \leftarrow a^2 \bmod n$
7 Return r

Pick a large prime number, ~200 digits long

p=

```
520617172688110225829850853409628900430579487996102545430889758440327219440458061993482627834790075028823780618996828085538214474401045588718257206979528176365692374670619236784807355507955499416644045699554849 9
```

$$(7, x, p) \rightarrow 7^x \bmod p$$

Remarkably easy

Assailing U.S. and Kiev, Putin Keeps Open Option of Force, By STEVEN LEE MYERS, ELLEN BARRY and ALAN COWELL 56 minutes ago, President Vladimir V. Putin of Russia on Tuesday described events in Ukraine as an unconstitutional coup, expressed contempt toward the United States and said that any potential use of military force would be a last resort.

```
sage:
power_mod(7,661615980609061103328546834632981100327606021844328118170610327602021315328013021032801317061210331202327112150001443267213283846986697832766969327789698283443269767669783266658282893298110032657665783267798769767632535433100611181702153298041144328115021606010211163287089801061006143286463281181706103312023283181616059733121032851802160098213301021600150599020033021902111715330610328608149806110132981532981033181100121116170617181706121098083300121812443302211315021616020033001211170210131633171219981500331705013286110617020032841698170215329811003315980600331704981632981121331312170211170598083318160133120233100609061698152133031215000133201218090032990132973308981616331502161215164 6,p)

4979297116885842175278633323006293290986549832694233577295952890262039411267933871586617511805097012352566334299626355536760020838695907697275222670277258353980892233568459367693160250707721036268518993463932429
```

$Y = 12345$

$$(7, p, Y) \longrightarrow x$$

such that $7^x \bmod p = Y$

Incredibly hard!

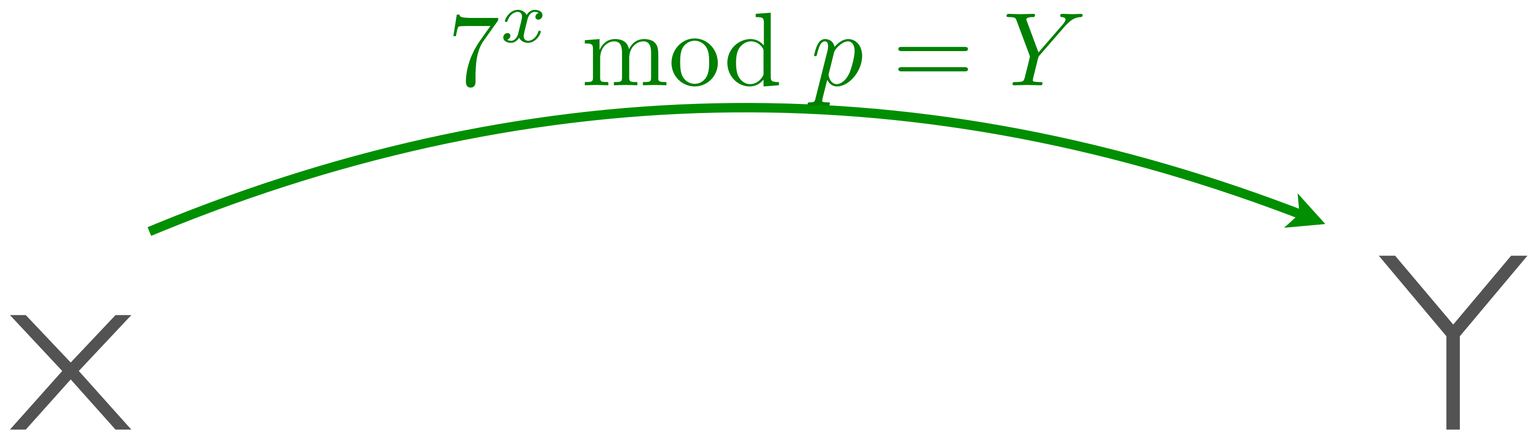
World record in discrete logarithms in GF(p)

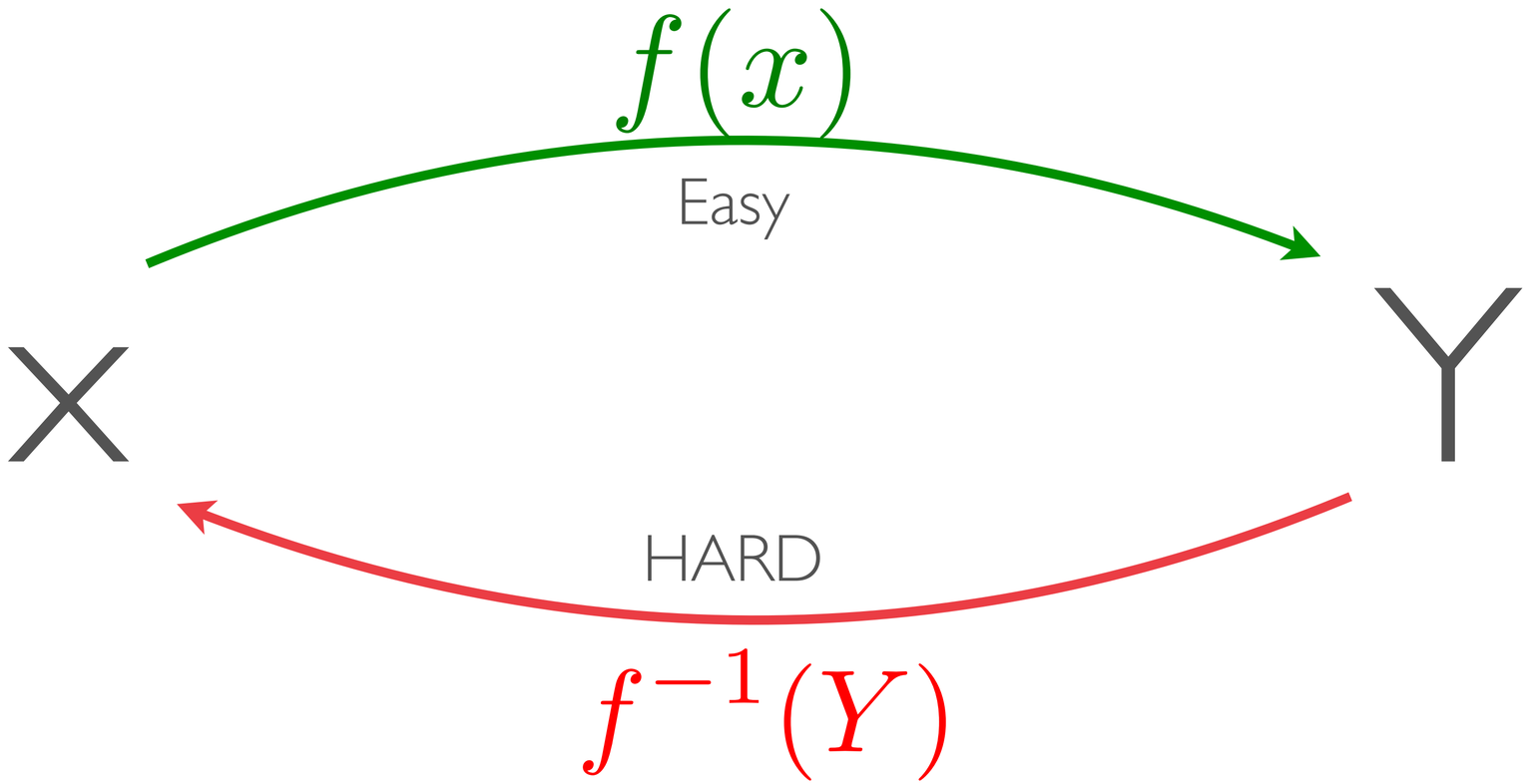
240 digits

3100yrs of CPU time

2 dec 2019

Fabrice Boudot, et al


$7^x \bmod p = Y$
X
Y



compute an x such that $7^x = Y \bmod p$



http://www.fitzmuseum.cam.ac.uk/gallery/chinesevases/sortingfragments.html







$f(x)$

Easy

X

Y

HARD

$f^{-1}(Y)$



http://fabpedigree.com/james/mathmen.htm

http://upload.wikimedia.org/wikipedia/commons/thumb/9/9b/Carl_Friedrich_Gauss.jpg/440px-Carl_Friedrich_Gauss.jpg

This “one way function” enables methods to build more principled pseudo-random generators.

Blum-Micali PRG

PRG(seed = S, g, p):

output msb(S)

$S \leftarrow g^S \bmod p$

msb_p(y) =

Example

$(g = 2,\ p = 13411)$

Suppose our seed was 722.

Blum–Micali PRG: a worked example

Parameters

p = 13411 (prime)

g = 2 (generates all of Z^*_p)

seed S_0 = 722

Each round

```
output msb(S)
S ← g^S mod p
```

msb(S) = 1 if S > (p−1)/2 = 6705
msb(S) = 0 otherwise

"Is S in the upper half?" — the hard-core bit of discrete log, not the binary leading bit.

i	S_i	S_i > 6705 ?	bit = msb(S_i)	next S = 2^S mod p
0	**722**	no	**0**	8111
1	**8111**	yes	**1**	4130
2	**4130**	no	**0**	8631
3	**8631**	yes	**1**	687
4	**687**	no	**0**	6484
5	**6484**	no	**0**	1681
6	**1681**	no	**0**	3441
7	**3441**	no	**0**	4395

Output stream (first 8 bits)

0	1	0	1	0	0	0	0

Toy parameters for teaching. Real use needs a prime of 2048+ bits.

Showing the work: $2^{722} \bmod 13411$

Square-and-multiply

Write the exponent in binary, then read it left to right:

- every bit: square
- bit = 1: also multiply by g = 2

$722 = 1011010010_2$

1	0	1	1	0	1	0	0	1	0

$2^{722} \equiv 8111 \pmod{13411}$

So $S_1 = 8111$: the “next S” in row 0.

10 squarings + 5 multiplications instead of 721 multiplications.

bit	square (mod p)	× 2 if bit = 1	running result
1	$1^2 = 1$	$1 \times 2 = 2$	**$2^1 = 2$**
0	$2^2 = 4$	—	**$2^2 = 4$**
1	$4^2 = 16$	$16 \times 2 = 32$	**$2^5 = 32$**
1	$32^2 = 1024$	$1024 \times 2 = 2048$	**$2^{11} = 2048$**
0	$2048^2 \equiv 10072$	—	**$2^{22} \equiv 10072$**
1	$10072^2 \equiv 4380$	$4380 \times 2 = 8760$	**$2^{45} \equiv 8760$**
0	$8760^2 \equiv 13269$	—	**$2^{90} \equiv 13269$**
0	$13269^2 \equiv 6753$	—	**$2^{180} \equiv 6753$**
1	$6753^2 \equiv 5609$	$5609 \times 2 = 11218$	**$2^{361} \equiv 11218$**
0	$11218^2 \equiv 8111$	—	**$2^{722} \equiv 8111$**

Toy parameters for teaching. Real use needs a prime of 2048+ bits.

Revisit our model for Encryption

Symmetric key enc has 1 major drawback.

Bob

Carol

Dave

Alice

Evan

George

Francis

Symmetric key enc has 1 major drawback.

$k_{ba}, k_{bc}, k_{bd}, k_{be}, k_{bf}, k_{bg}$

Bob

$k_{ca}, k_{cb}, k_{cd}, k_{ce}, k_{cf}, k_{cg}$

Carol

$k_{da}, k_{db}, k_{dc}, k_{de}, k_{df}, k_{dg}$

Dave

Alice

$k_{ab}, k_{ac}, k_{ad}, k_{ae}, k_{af}, k_{ag}$

$O(n^2)$ keys to manage!

Evan

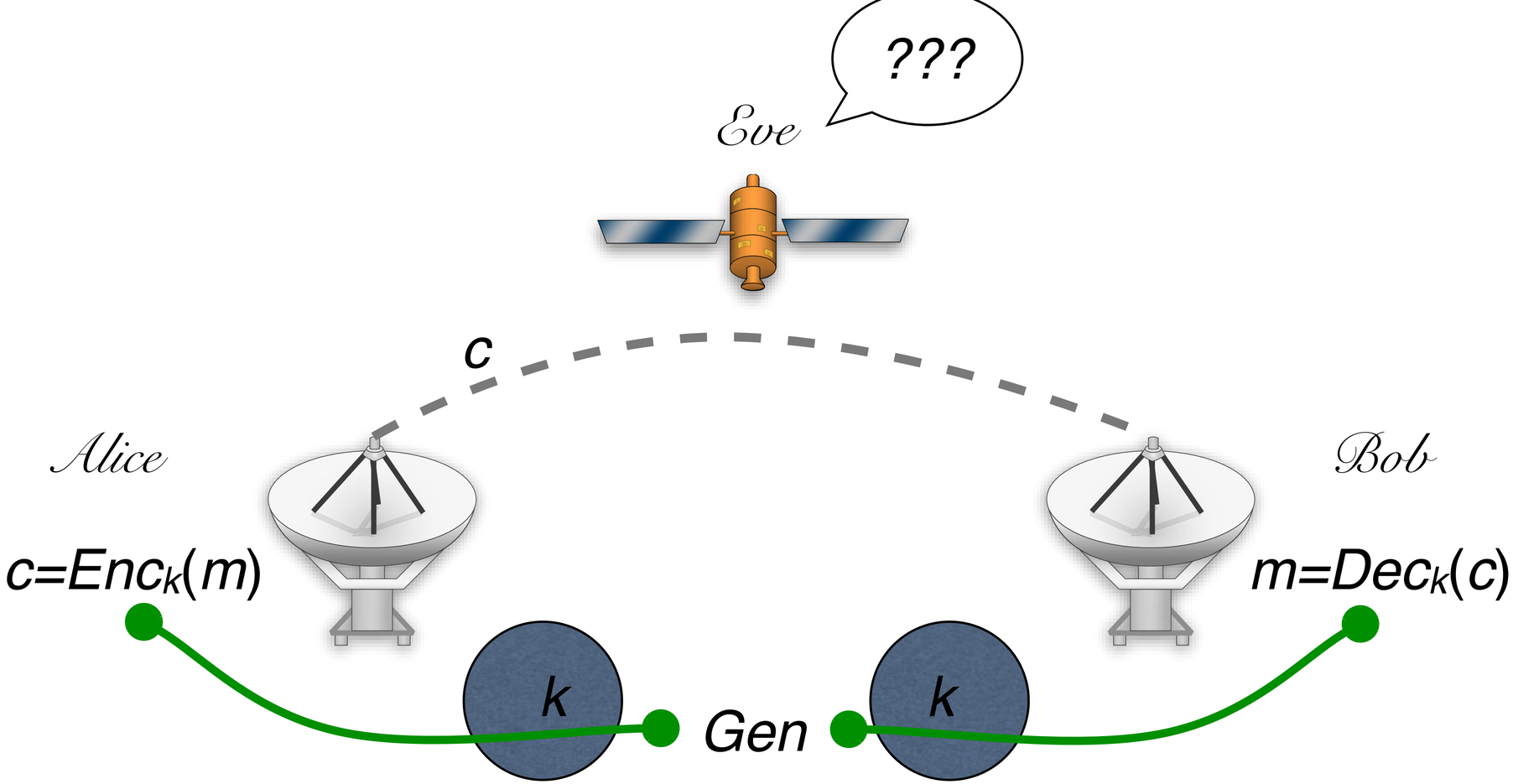
$k_{ea}, k_{eb}, k_{ec}, k_{ed}, k_{ef}, k_{eg}$

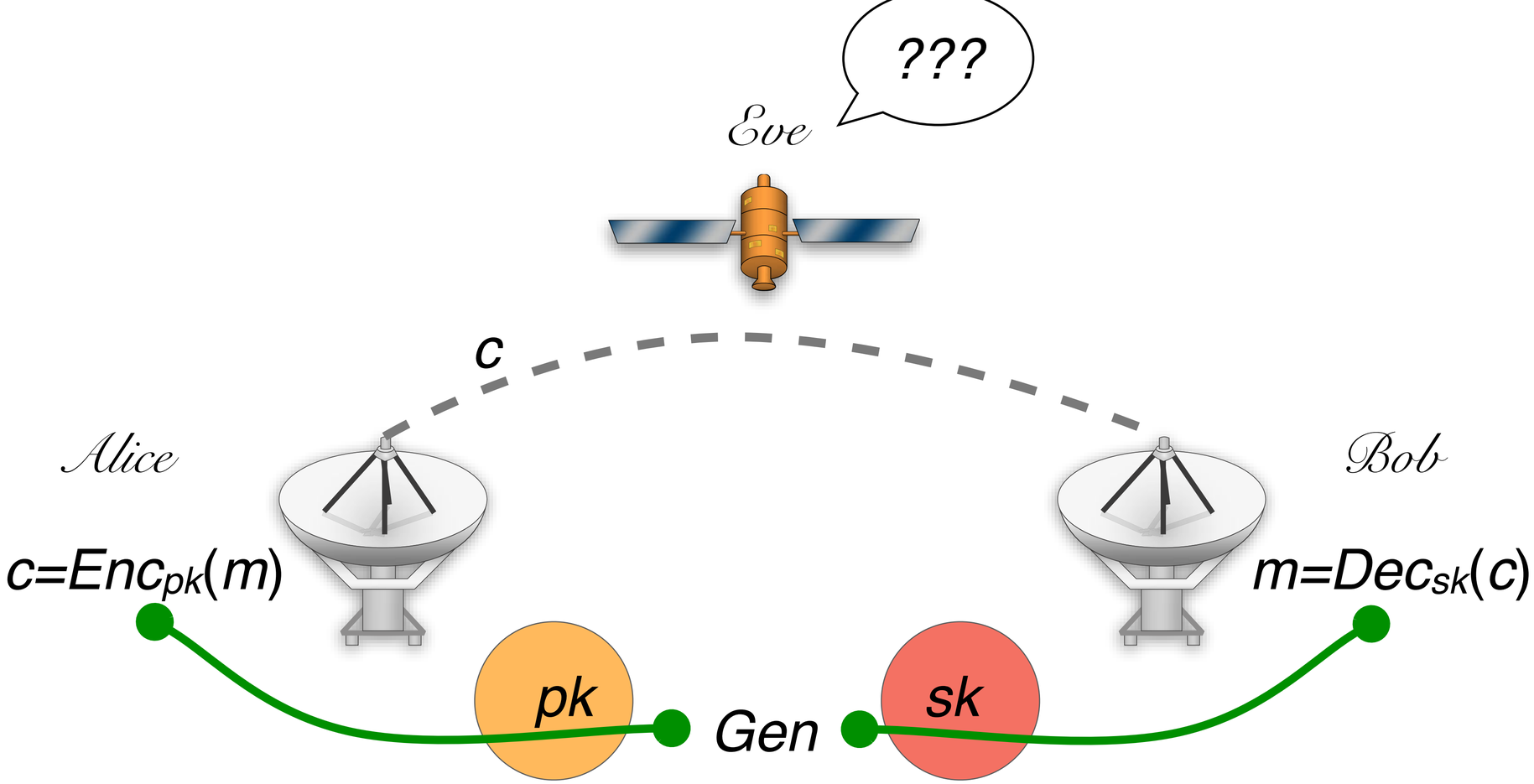
George

$k_{ga}, k_{gb}, k_{gc}, k_{gd}, k_{ge}, k_{gf}$

Francis

$k_{fa}, k_{fb}, k_{fc}, k_{fd}, k_{fe}, k_{fg}$


???
Eve
c
Alice
Bob
c=Enc_k(m)
m=Dec_k(c)
k
Gen
k



Pk can be used to encrypt.

sk can be used to decrypt.

PKC key enc

sk_b

Bob

sk_c

Carol

sk_d

Dave

Alice

sk_a

$pk_a, pk_b, pk_c, pk_d, pk_e, pk_f, pk_g$

Are publicly posted

Evan

sk_e

George

sk_g

Francis

sk_f

Public key encryption

Gen Enc Dec

3 algorithms

Gen (key generation)

$(pk, sk) \leftarrow \mathsf{Gen}(1^n)$

Enc (encryption)

$c \leftarrow \mathsf{Enc}_{pk}(m)$ for $pk \in \mathcal{K}, m \in \mathcal{M}$

Dec (decryption)

Public key encryption

Gen Enc Dec

3 algorithms

Gen (key generation)

$$(pk, sk) \leftarrow \mathsf{Gen}(1^n)$$

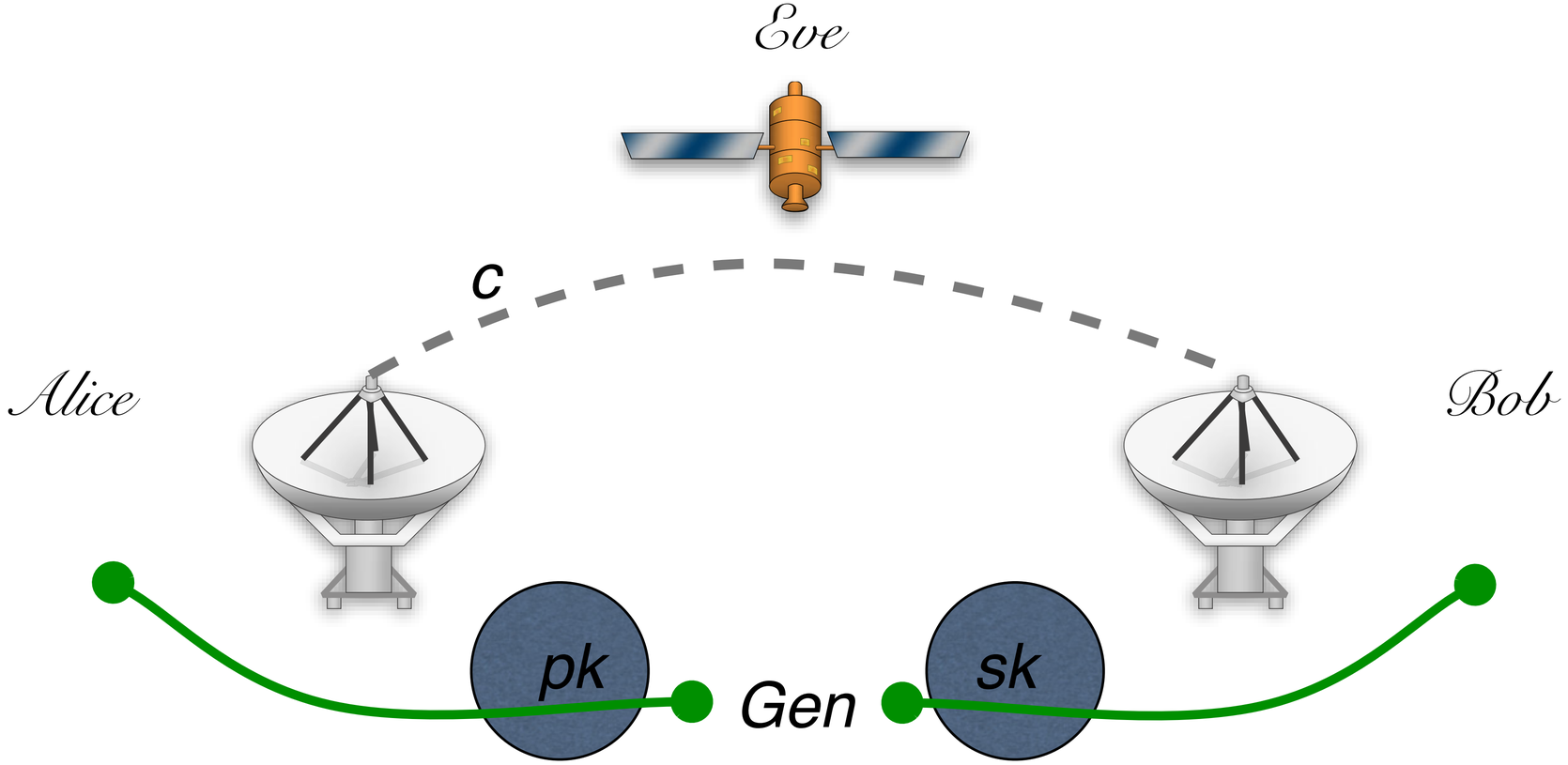
Enc (encryption)

$$c \leftarrow \mathsf{Enc}_{pk}(m) \text{ for } pk \in \mathcal{K}, m \in \mathcal{M}$$

Dec (decryption)

$$\forall m \in \mathcal{M}, (pk, sk) \leftarrow \mathsf{Gen}(1^n)$$

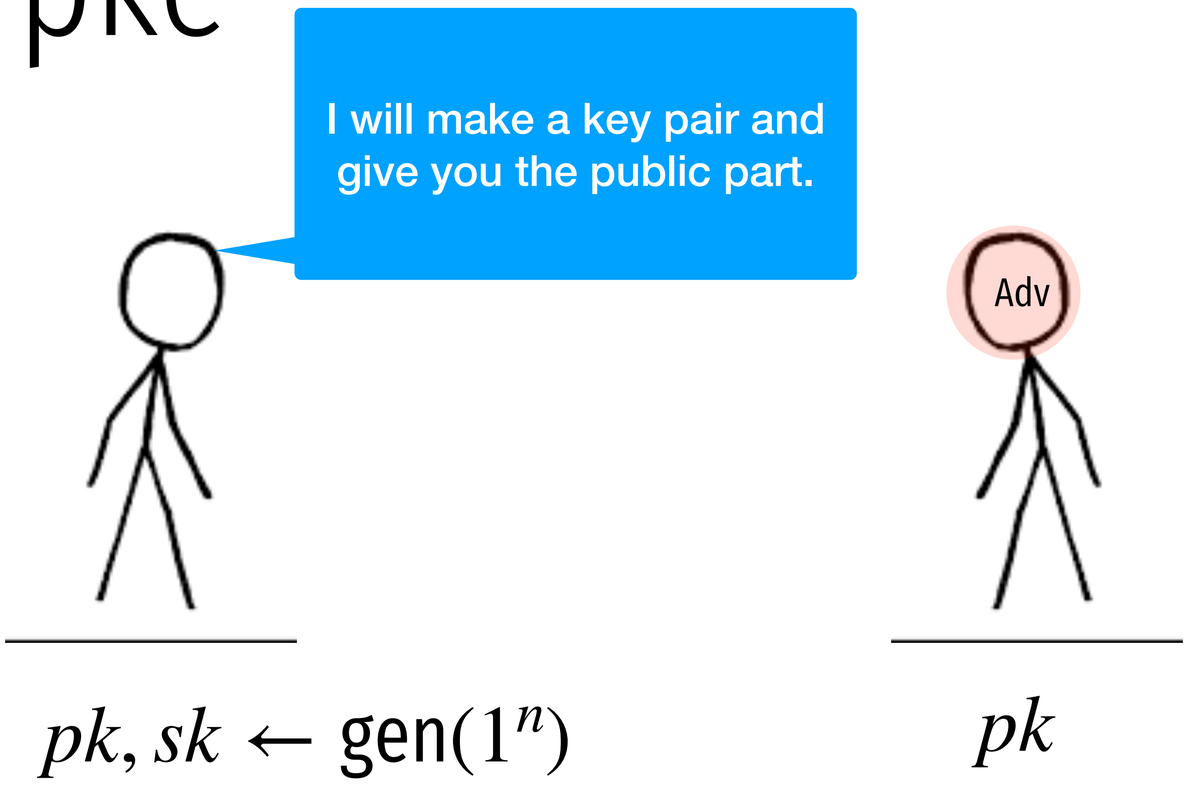
$$\Pr[\mathsf{Dec}_{sk}(\mathsf{Enc}_{pk}(m)) = m] = 1$$



"for any pair of messages m_1, m_2,
Eve cannot tell whether $c = Enc_{pk}(m_i)$."

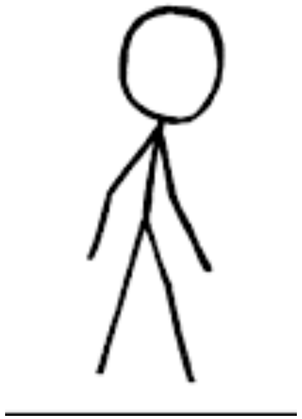
IND-CPA security for pke

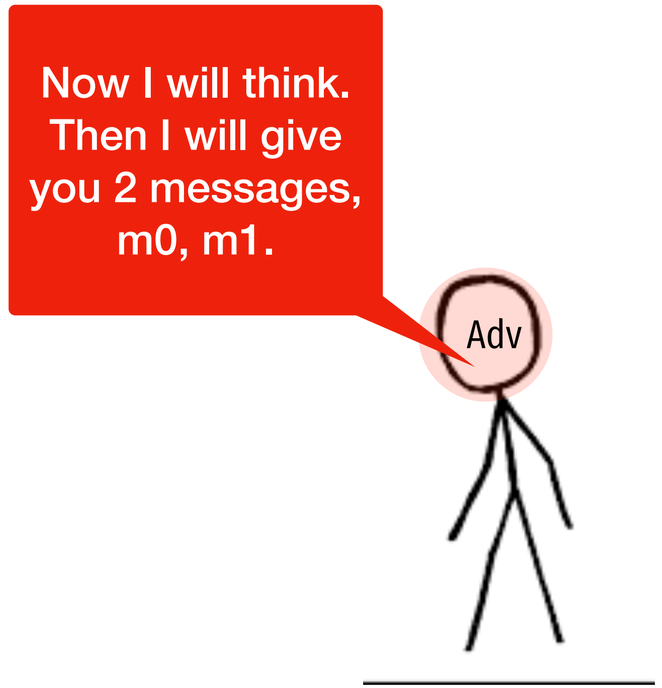
(weakest notion of security)



IND-CPA security for pke

(weakest notion of security)





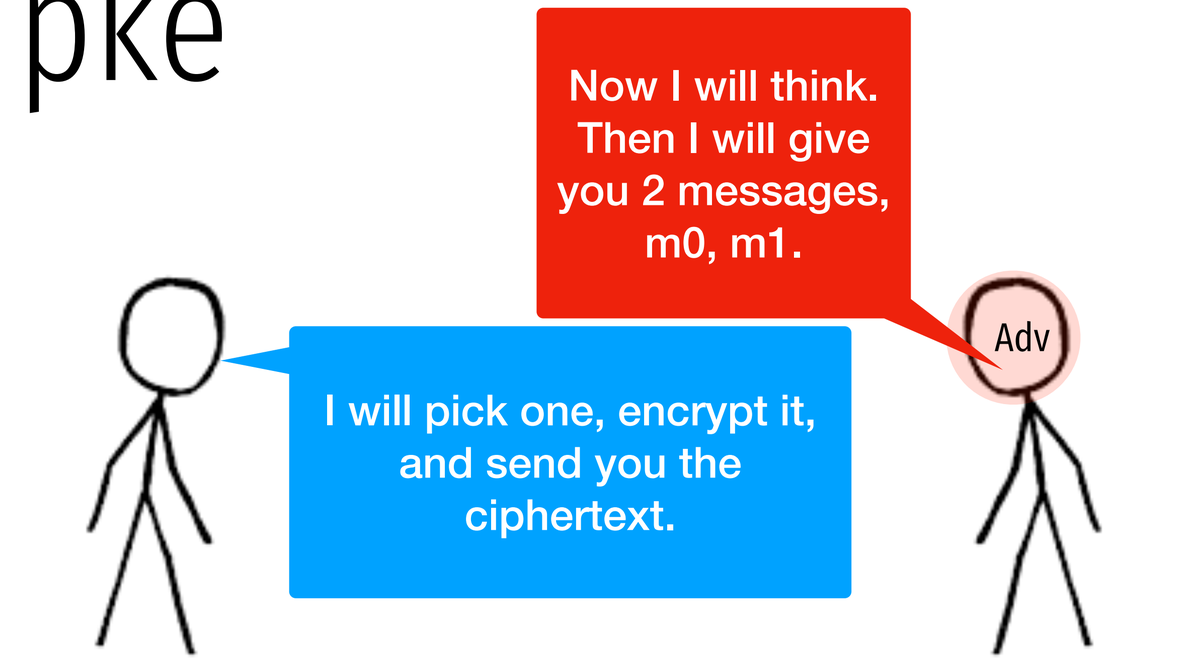
$pk, sk \leftarrow \mathsf{gen}(1^n)$

pk

$m_0, m_1 \leftarrow A(pk)$

IND-CPA security for pke

(weakest notion of security)



$pk, sk \leftarrow \mathsf{gen}(1^n)$

pk

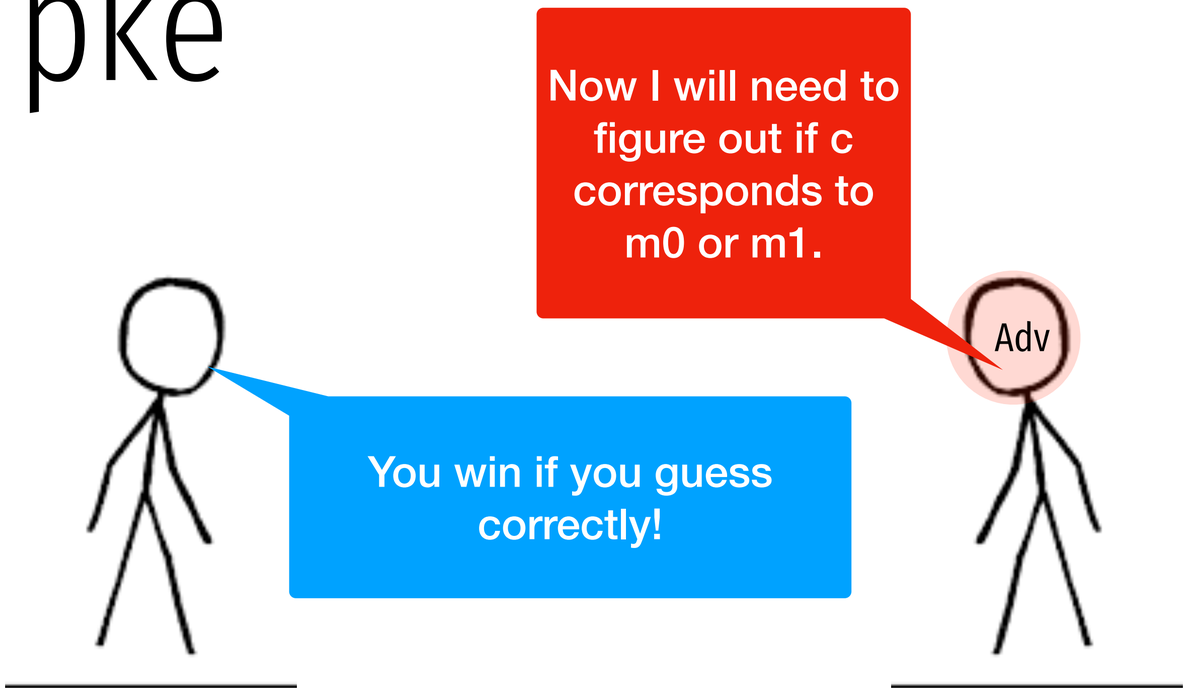
$m_0, m_1 \leftarrow A(pk)$

$b \leftarrow \{0,1\}$

$c \leftarrow \mathsf{enc}_{pk}(m_b)$

IND-CPA security for pke

(weakest notion of security)



$pk, sk \leftarrow \mathsf{gen}(1^n)$

pk

$m_0, m_1 \leftarrow A(pk)$

$b \leftarrow \{0,1\}$

$c \leftarrow \mathsf{enc}_{pk}(m_b)$

$b' \leftarrow A(pk, m_0, m_1, c)$

IND-CPA security for pke

(weakest notion of security)

$$pk, sk \leftarrow \mathsf{gen}(1^n)$$

$$m_0, m_1 \leftarrow A(pk)$$

$$b \leftarrow \{0,1\}$$

$$c \leftarrow \mathsf{enc}_{pk}(m_b)$$

$$b' \leftarrow A(pk, m_0, m_1, c)$$

$$\Pr[b = b'] = 1/2 + \epsilon(n)$$

How to build public key encryption?

Basic Number theory

Modular Exponentiation

$(a, x, n) \rightarrow a^x \bmod n$

$7^{19} \bmod 31$

$$\underbrace{7 \cdot 7 \cdots 7}_{19 \text{ times}}$$

Modular Exponentiation

$(a, x, n) \rightarrow a^x \bmod n$

$7^{19} \bmod 31$

Modular Exponentiation

$(a, x, n) \rightarrow a^x \bmod n$

$7^{19} \bmod 31$

Modular Exponentiation $(a, x, n) \rightarrow a^x \bmod n$

$7^{19} \bmod 31$

$7^1 \quad 7^2 \quad 7^4 \quad 7^8 \quad 7^{16} \pmod{31}$

Modular Exponentiation

$(a, x, n) \rightarrow a^x \bmod n$

$$7^{19} \bmod 31$$

$$7^1 \quad 7^2 \quad 7^4 \quad 7^8 \quad 7^{16} \pmod{31}$$

7 18 14 10 7

Modular Exponentiation

$$(a, x, n) \rightarrow a^x \bmod n$$

Algorithm 2: ModularExponentiation(a, x, n)

Input: $a, x \in [1, n]$

$r \leftarrow 1$

while $x > 0$ **do**

  if x *is odd* **then**

    $r \leftarrow r \cdot a \bmod n$

  $x \leftarrow \lfloor x/2 \rfloor$

  $a \leftarrow a^2 \bmod n$

Return r

Modular Exponentiation

$$(a, x, n) \rightarrow a^x \bmod n$$

$$a^x \bmod n = \prod_{i=0}^{\ell} x_i a^{2^i} \bmod n$$

Algorithm 2: ModularExponentiation(a, x, n)

Input: $a, x \in [1, n]$

1 $r \leftarrow 1$
2 **while** $x > 0$ **do**
3 **if** x *is odd* **then**
4 $r \leftarrow r \cdot a \bmod n$
5 $x \leftarrow \lfloor x/2 \rfloor$
6 $a \leftarrow a^2 \bmod n$
7 Return r

Greatest Common Divisor

GCD(A,B) = GCD( )

Greatest Common Divisor

$$GCD(A,B) = GCD(B, A \bmod B)$$

Greatest Common Divisor

GCD(6809,1641)

Greatest Common Divisor

`GCD(6809,1641)`

GCD(1641, 245)

GCD(245, 171)

GCD(171, 74)

GCD(74, 23)

GCD(23, 5)

GCD(5, 3)

GCD(3, 2)

GCD(2, 1)

$6809 = 4 \cdot 1641 + 245$

$1641 = 6 \cdot 245 + 171$

$245 = 1 \cdot 171 + 74$

$171 = 2 \cdot 74 + 23$

$74 = 3 \cdot 23 + 5$

$23 = 4 \cdot 5 + 3$

$5 = 1 \cdot 3 + 2$

$3 = 1 \cdot 2 + 1$

$2 = 2 \cdot 1 + 0$

given (a,b), finds (x,y) s.t.
ax + by = gcd(a,b)

Algorithm 1: ExtendedEuclid(a, b)

Input: (a, b) s.t $a > b \geq 0$
Output: (x, y) s.t. $ax + by = \gcd(a, b)$

```
if a mod b = 0 then
    Return (0, 1)
else
    (x, y) ← ExtendedEuclid (b, a mod b)
    Return (y, x − y(⌊a/b⌋))
```

Greatest Common Divisor

GCD(6809,1641)

GCD(1641, 245)

GCD(245, 171)

GCD(171, 74)

GCD(74, 23)

GCD(23, 5)

GCD(5, 3)

GCD(3, 2) (1, 0 - 1*1)

GCD(2, 1) (0,1)

$6809 = 4 \cdot 1641 + 245$ (-643, 2668)

$1641 = 6 \cdot 245 + 171$ (96, -643

$245 = 1 \cdot 171 + 74$ (-67, 96)

$171 = 2 \cdot 74 + 23$ (29, -67)

$74 = 3 \cdot 23 + 5$ (-9, 29)

$23 = 4 \cdot 5 + 3$ (2, -9)

$5 = 1 \cdot 3 + 2$ (-1, 2)

$3 = 1 \cdot 2 + 1$ (1, -1)

$2 = 2 \cdot 1 + 0$ (0, 1)

Greatest Common Divisor

GCD(6809,1641)

GCD(1641, 245)

GCD(245, 171)

GCD(171, 74)

GCD(74, 23)

GCD(23, 5)

GCD(5, 3)

GCD(3, 2)

GCD(2, 1)

(0,1)

$6809 = 4 \cdot 1641 + 245$

$1641 = 6 \cdot 245 + 171$

$245 = 1 \cdot 171 + 74$

$171 = 2 \cdot 74 + 23$

$74 = 3 \cdot 23 + 5$

$23 = 4 \cdot 5 + 3$

$5 = 1 \cdot 3 + 2$

$3 = 1 \cdot 2 + 1$

$2 = 2 \cdot 1 + 0$

Greatest Common Divisor

GCD(6809,1641)

GCD(1641, 245)

GCD(245, 171)

GCD(171, 74)

GCD(74, 23)

GCD(23, 5)

GCD(5, 3)

GCD(3, 2) (1, 0 - 1*1)

GCD(2, 1) (0,1)

$6809 = 4 \cdot 1641 + 245$

$1641 = 6 \cdot 245 + 171$

$245 = 1 \cdot 171 + 74$

$171 = 2 \cdot 74 + 23$

$74 = 3 \cdot 23 + 5$

$23 = 4 \cdot 5 + 3$

$5 = 1 \cdot 3 + 2$

$3 = 1 \cdot 2 + 1$

$2 = 2 \cdot 1 + 0$

Greatest Common Divisor

GCD(6809,1641)

GCD(1641, 245)

GCD(245, 171)

GCD(171, 74)

GCD(74, 23)

GCD(23, 5)

GCD(5, 3)

GCD(3, 2)

GCD(2, 1)

(1,-1)

(0,1)

$6809 = 4 \cdot 1641 + 245$

$1641 = 6 \cdot 245 + 171$

$245 = 1 \cdot 171 + 74$

$171 = 2 \cdot 74 + 23$

$74 = 3 \cdot 23 + 5$

$23 = 4 \cdot 5 + 3$

$5 = 1 \cdot 3 + 2$

$3 = 1 \cdot 2 + 1$

$2 = 2 \cdot 1 + 0$

Greatest Common Divisor

`GCD(6809,1641)`

GCD(1641, 245)

GCD(245, 171)

GCD(171, 74)

GCD(74, 23)

GCD(23, 5)

GCD(5, 3)

GCD(3, 2)

GCD(2, 1)

`(-1,2)`

`(1,-1)`

`(0,1)`

$6809 = 4 \cdot 1641 + 245$

$1641 = 6 \cdot 245 + 171$

$245 = 1 \cdot 171 + 74$

$171 = 2 \cdot 74 + 23$

$74 = 3 \cdot 23 + 5$

$23 = 4 \cdot 5 + 3$

$5 = 1 \cdot 3 + 2$

$3 = 1 \cdot 2 + 1$

$2 = 2 \cdot 1 + 0$

Greatest Common Divisor

GCD(6809,1641)

GCD(1641, 245)	
GCD(245, 171)	
GCD(171, 74)	
GCD(74, 23)	
GCD(23, 5)	(2,-9)
GCD(5, 3)	(-1,2)
GCD(3, 2)	(1,-1)
GCD(2, 1)	(0,1)

$6809 = 4 \cdot 1641 + 245$ $(-643, 2668)$

$1641 = 6 \cdot 245 + 171$ $(96, -643)$

$245 = 1 \cdot 171 + 74$ $(-67, 96)$

$171 = 2 \cdot 74 + 23$ $(29, -67)$

$74 = 3 \cdot 23 + 5$ $(-9, 29)$

$23 = 4 \cdot 5 + 3$ $(2, -9)$

$5 = 1 \cdot 3 + 2$ $(-1, 2)$

$3 = 1 \cdot 2 + 1$ $(1, -1)$

$2 = 2 \cdot 1 + 0$ $(0, 1)$

Greatest Common Divisor

GCD(6809,1641)

GCD allows us to compute modular inverses.

6809*(-643) + 1641*2668 =

Greatest Common Divisor

GCD(6809,1641)

```
6809*(-643) + 1641*2668 =   1

  -437818       437818
```

GCD allows us to compute modular inverses.

Euler totient



Euler totient

$\phi(15) =$

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

Euler totient

prime

$$\Phi(p) = p - 1$$

product of 2 primes

$$\Phi(n) = (p - 1)(q - 1)$$

Example of groups

$$(\mathbb{Z}_n, \star) \qquad \{a \mid \gcd(a, n) = 1\} \qquad \mathbb{Z}_n^\star$$

multiplicative group, mod n

Z*_15 = {1,2,4,7,8,11,13,14}

$$|\mathbb{Z}_n^\star| = \Phi(n)$$

Euler theorem

$$\forall a \in \mathbb{Z}_n^\star,\ a^{\Phi(n)} = 1 \bmod n$$

Examples

$$7^{30} \bmod 31 =$$

1	2	4	8	16
7	18	14	10	7

Examples

$$2^8 \bmod 15 =$$

Implications of Euler

$$a^{10\phi(N)} \bmod N =$$

$$a^{k\phi(N)+1} \bmod N =$$

compute

$$11^{30^{2021}} \bmod 23$$

(show your work)

“Textbook” RSA (insecure)

Pick N = p*q where p,q are primes.

“Textbook” RSA (insecure)

Pick N = p*q where p,q are primes.

Pick e,d such that $e \cdot d = 1 \mod \phi(N)$

“Textbook” RSA (insecure)

Pick N = p*q where p,q are primes.

Pick e,d such that $e \cdot d = 1 \mod \phi(N)$

$\text{Enc}_{N,e}(m) = m^e \mod N$

$\text{Dec}_{N,d}(c) = c^d \mod N$

“Textbook” RSA (insecure)

Pick N = p*q where p,q are primes.

Pick e,d such that $e \cdot d = 1 \mod \phi(N)$

$\text{Enc}_{N,e}(m) = m^e \mod N$

$\text{Dec}_{N,d}(c) = c^d \mod N$

$(m^e)^d \mod N =$

Example of Textbook RSA

m=5

PK = (N=143, e=7) SK = (d=103)

“Textbook” RSA (insecure)

Pick N = p*q where p,q are primes.

Pick e,d such that $e \cdot d = 1 \mod \phi(N)$

$\text{Enc}_{N,e}(m) = m^e \mod N$

$\text{Dec}_{N,d}(c) = c^d \mod N$

Why is it insecure against IND-CPA attack?

pkcs1.5

ENC$_{pk}(m)$

PICK r AS A RANDOM STRING WITH NO 0s

(TYPICALLY 8 BYTES)

$$c \leftarrow (0||2||r||0||m)^e \bmod N$$

"PADDING ORACLE" ATTACK AGAINST THIS SCHEME

RSA-OAEP+

$\textsc{Gen}(1^n)$

$f, f^{-1} \leftarrow \textsc{trapdoor owp}()$

$\textsc{Enc}_{pk}(m)$

$r \leftarrow U_n$

$s \leftarrow R_1(r) \oplus m \;||\; R_2(r||m)$

$t \leftarrow R_3(s) \oplus r$

$c \leftarrow f(s||t)$

$R_1 : \{0,1\}^{k_0} \rightarrow \{0,1\}^n$

$R_2 : \{0,1\}^{n+k_0} \rightarrow \{0,1\}^{k_1}$

$R_3 : \{0,1\}^{n+k_1} \rightarrow \{0,1\}^{k_0}$

$\textsc{Dec}_{sk}(c)$

$(s = (s_1, s_2), t) \leftarrow f^{-1}(c)$

$r \leftarrow R_3(s) \oplus t$

$m \leftarrow R_1(r) \oplus s_1$

$R_2(r||m) \stackrel{?}{=} s_2$ $\quad$ $\textsc{output } m \textsc{ else fail}$

Example: apple.com

www.apple.com | Apple Public EV Server RSA CA 2 - G1 | DigiCert High Assurance EV Root CA

Subject Name

Business Category	Private Organization
Inc. Country	US
Inc. State/Province	California
Serial Number	C0806592
Country	US
State/Province	California
Locality	Cupertino
Organization	Apple Inc.
Common Name	www.apple.com

Issuer Name

Country	US
Organization	Apple Inc.
Common Name	Apple Public EV Server RSA CA 2 - G1

Validity

Not Before	Fri, 23 Aug 2024 17:30:11 GMT
Not After	Thu, 21 Nov 2024 17:40:11 GMT

Subject Alt Names

DNS Name	www.apple.com
DNS Name	images.apple.com
DNS Name	www.apple.com.cn

Public Key Info

Algorithm	RSA
Key Size	2048
Exponent	65537
Modulus	C8:A2:02:8A:C1:1C:A7:9A:EE:58:49:9B:10:3C:41:8D:BF:EF:6F:23:7E:64:05...

Safari is using an encrypted connection to www.apple.com.

Encryption with a digital certificate keeps information private as it's sent to or from the https website www.apple.com.

DigiCert, Inc. has identified www.apple.com as being owned by Apple Inc. in Cupertino, California, US.

DigiCert High Assurance EV Root CA
└ DigiCert SHA2 Extended Validation Server CA-3
└ www.apple.com

Serial Number	03 8E 3F 9E 09 D7 ED C7 B1 80 3F 74 A7 4C 35 AB
Version	3
Signature Algorithm	SHA-256 with RSA Encryption (1.2.840.113549.1.1.11)
Parameters	None
Not Valid Before	Tuesday, October 6, 2020 at 8:00:00 PM Eastern Daylight Time
Not Valid After	Friday, October 8, 2021 at 8:00:00 AM Eastern Daylight Time
Public Key Info	
Algorithm	RSA Encryption (1.2.840.113549.1.1.1)
Parameters	None
Public Key	256 bytes : CA 1B 1C 21 78 15 3D 40 CF A3 79 3F 9D CF B2 53 AB A9 41 FF 3E 06 A1 29 69 8A 04 46 9E FB C4 0D 56 7A CA E6 80 E7 AF C6 C0 BF 8B 60 71 CA 9A E8 76 0C 06 C8 9B 77 B8 F3 1B EA 7E E7 3A 84 CB A3 88 A5 93 04 3F 69 66 77 CF AE 06 D1 D9 E1 10 08 7A E0 24 98 E7 56 97 0F 73 68 7B 4D 69 46 28 26 FF 05 81 0C C0 DA FC 21 71 81 65 9A 39 C9 E9 68 36 36 02 5F 81 80 B7 7E 8A 5B FE 34 D0 CE 76 2D D9 8B 3E D4 13 C0 EC EB 0F 2C 77 AD 1E 7B 20 F6 DA 92 98 FD 89 F3 A7 CB 53 16 2E B0 B9 62 BE C8 C3 28 40 CF 8C 5C 61 77 8F 92 3D 2F 23 F2 0A AB 65 82 22 B8 98 CE BA C8 00 95 E4 67 34 6E 76 E5 D1 D3 2D 51 91 BC EF C0 C8 DE F8 7B CC 46 45 00 76 D9 CB 30 31 E9 56 FD 0E 68 F4 36 F9 1B 5F 88 61 62 8F 60 A8 DE 43 7B 5C C1 15 73 D4 06 12 6E 85 9B 50 9C 24 BF 5F FC F4 68 95 67 D5 BF 44 71
Exponent	65537
Key Size	2,048 bits

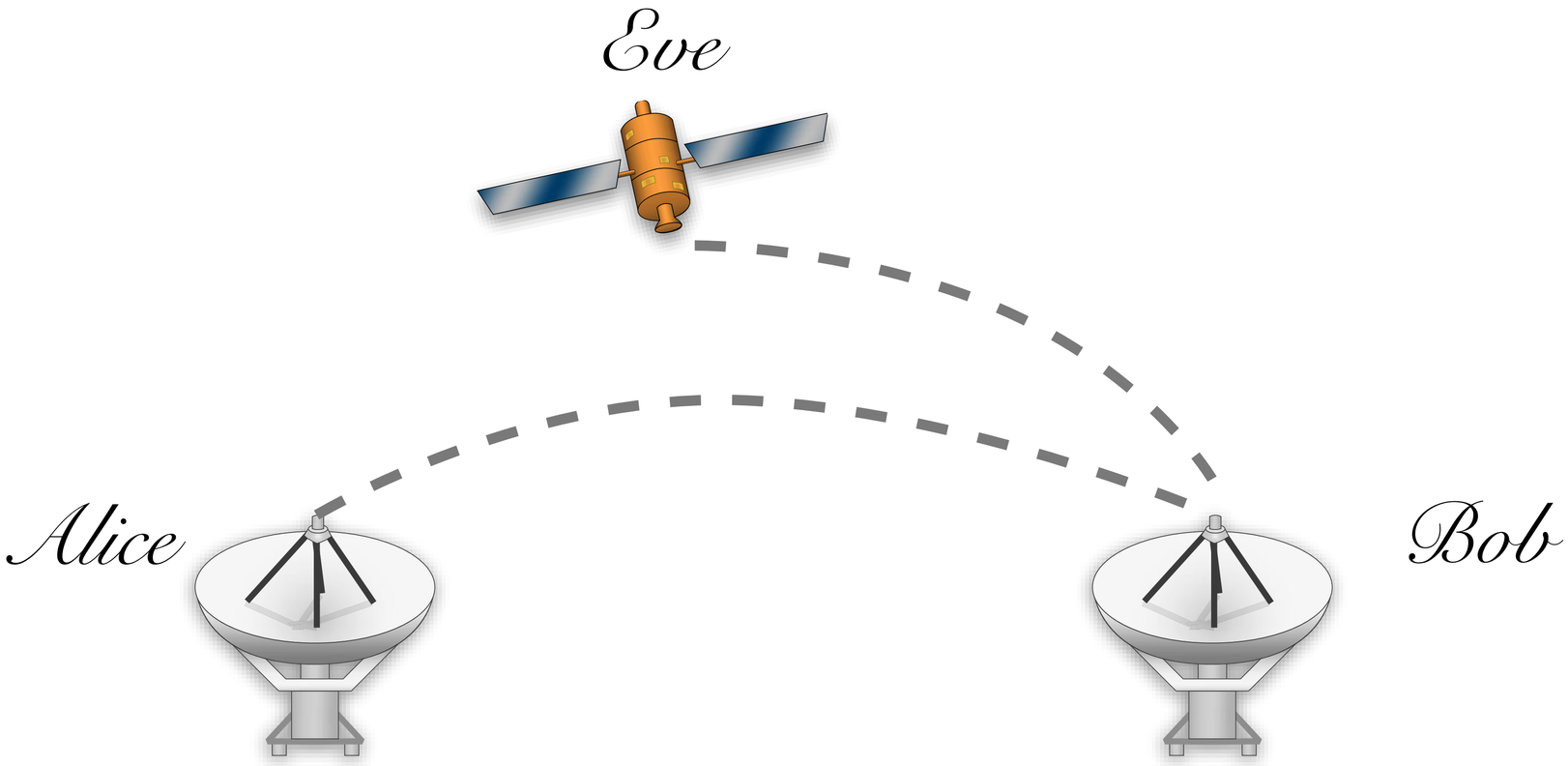
? Hide Certificate OK

Very old problem

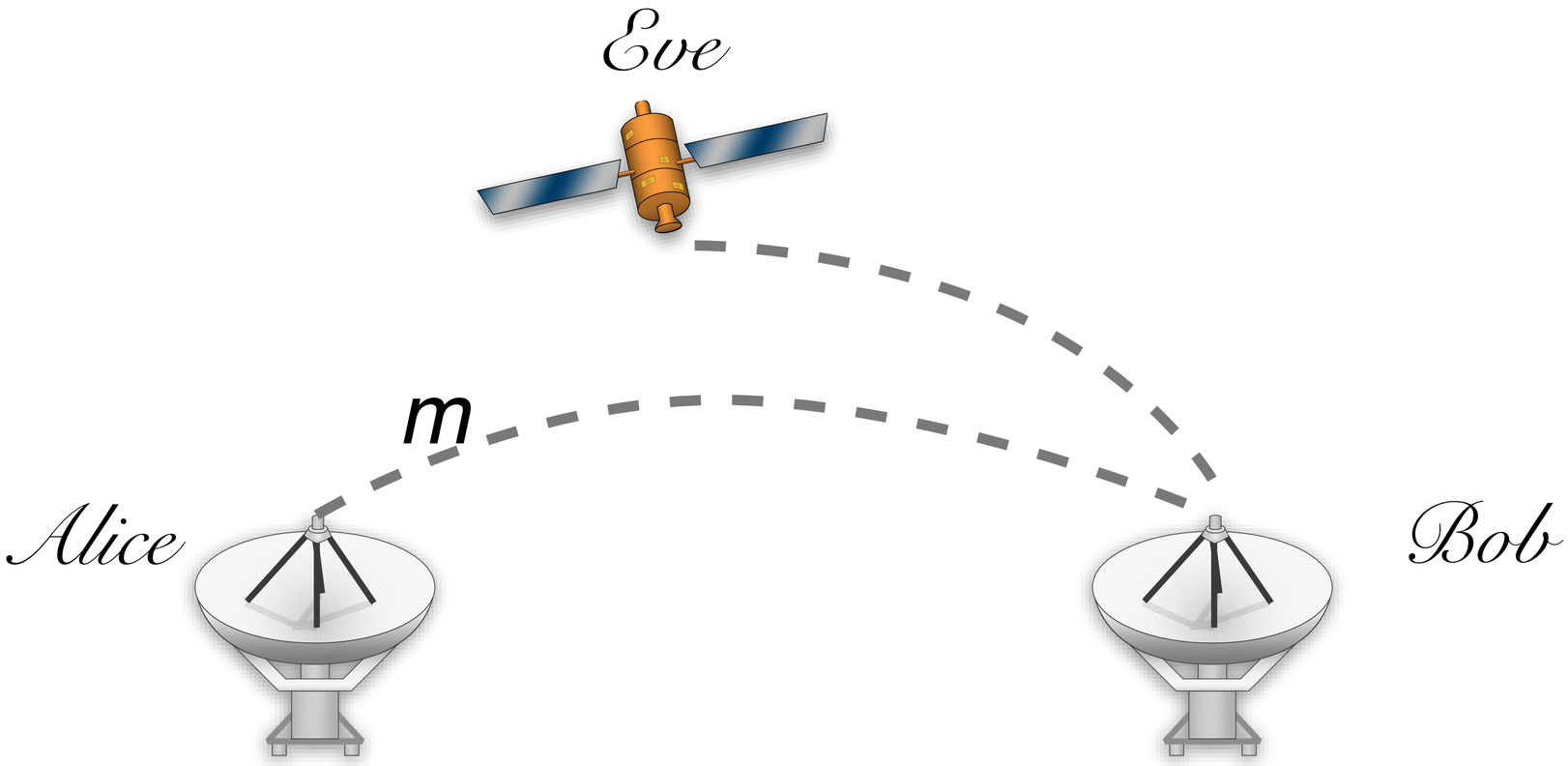




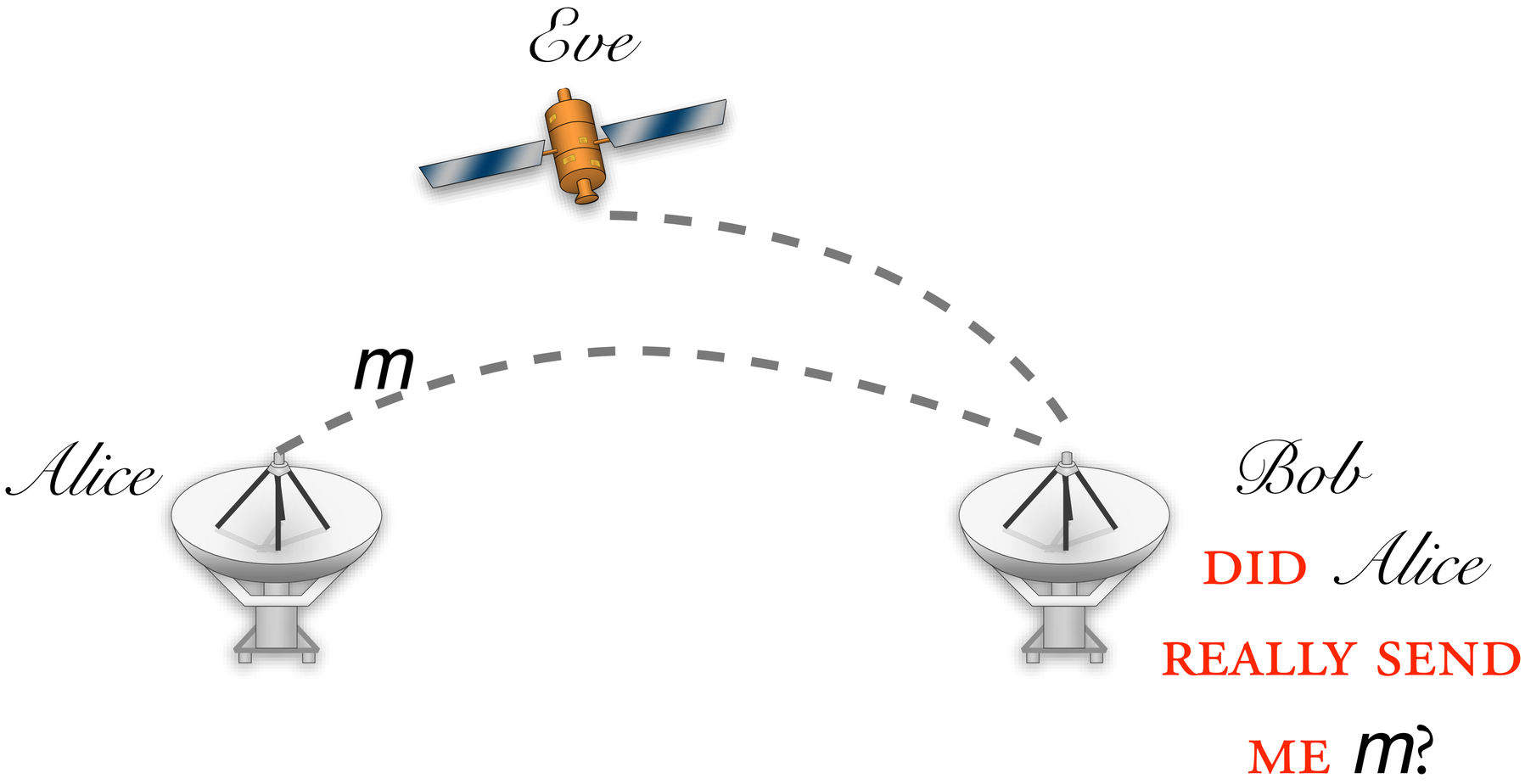
New Problem



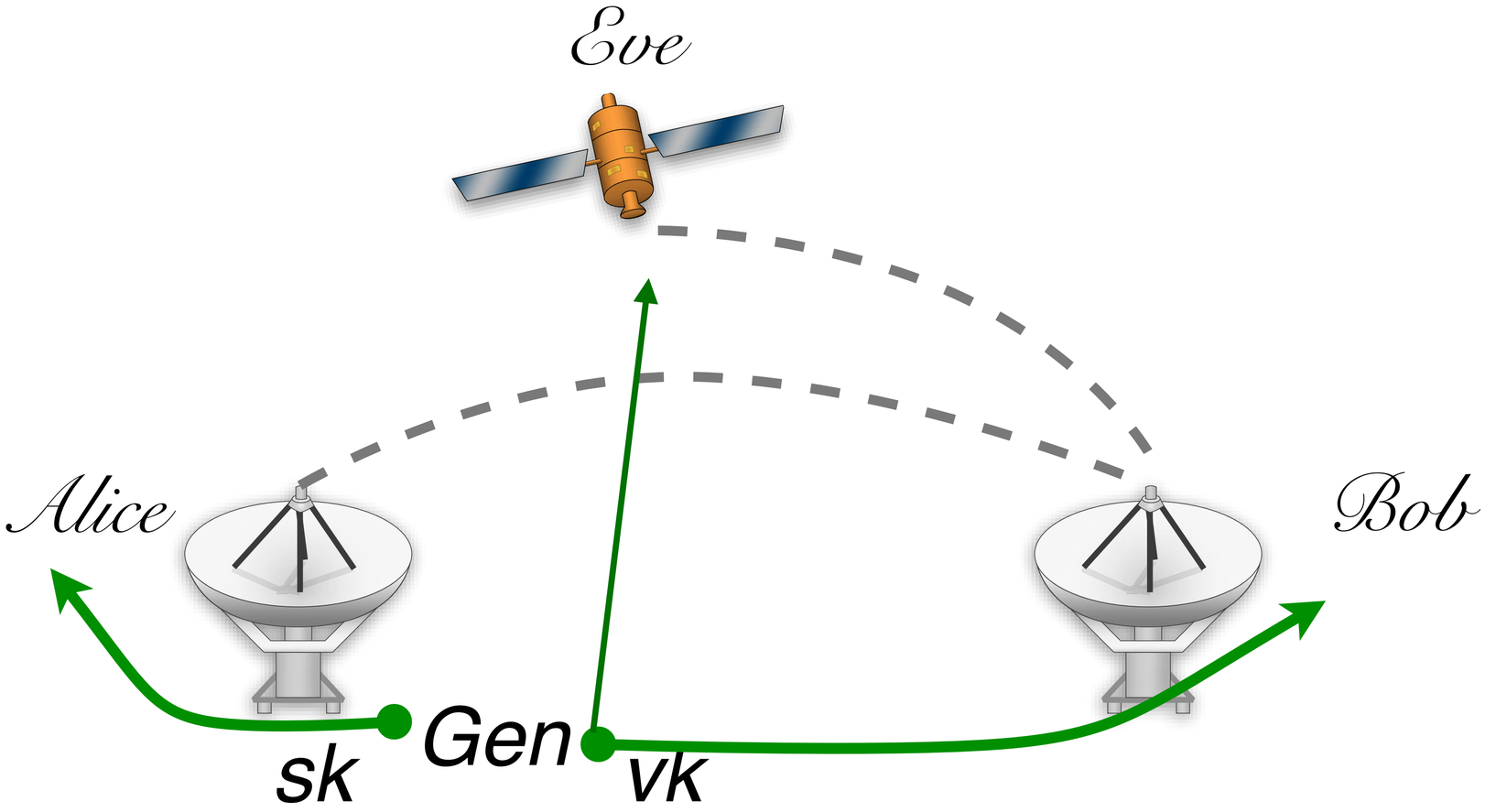
New Problem



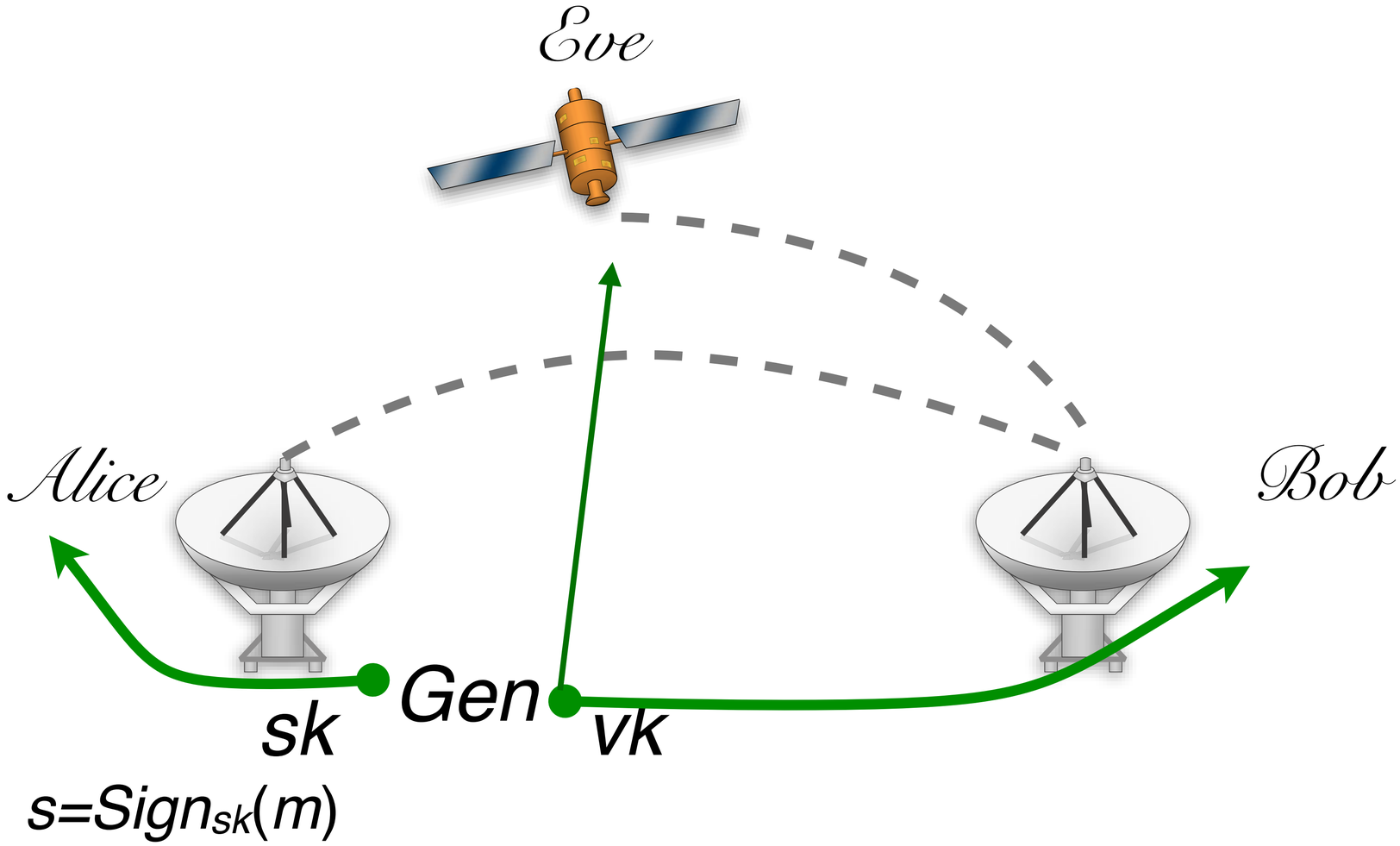
New Problem



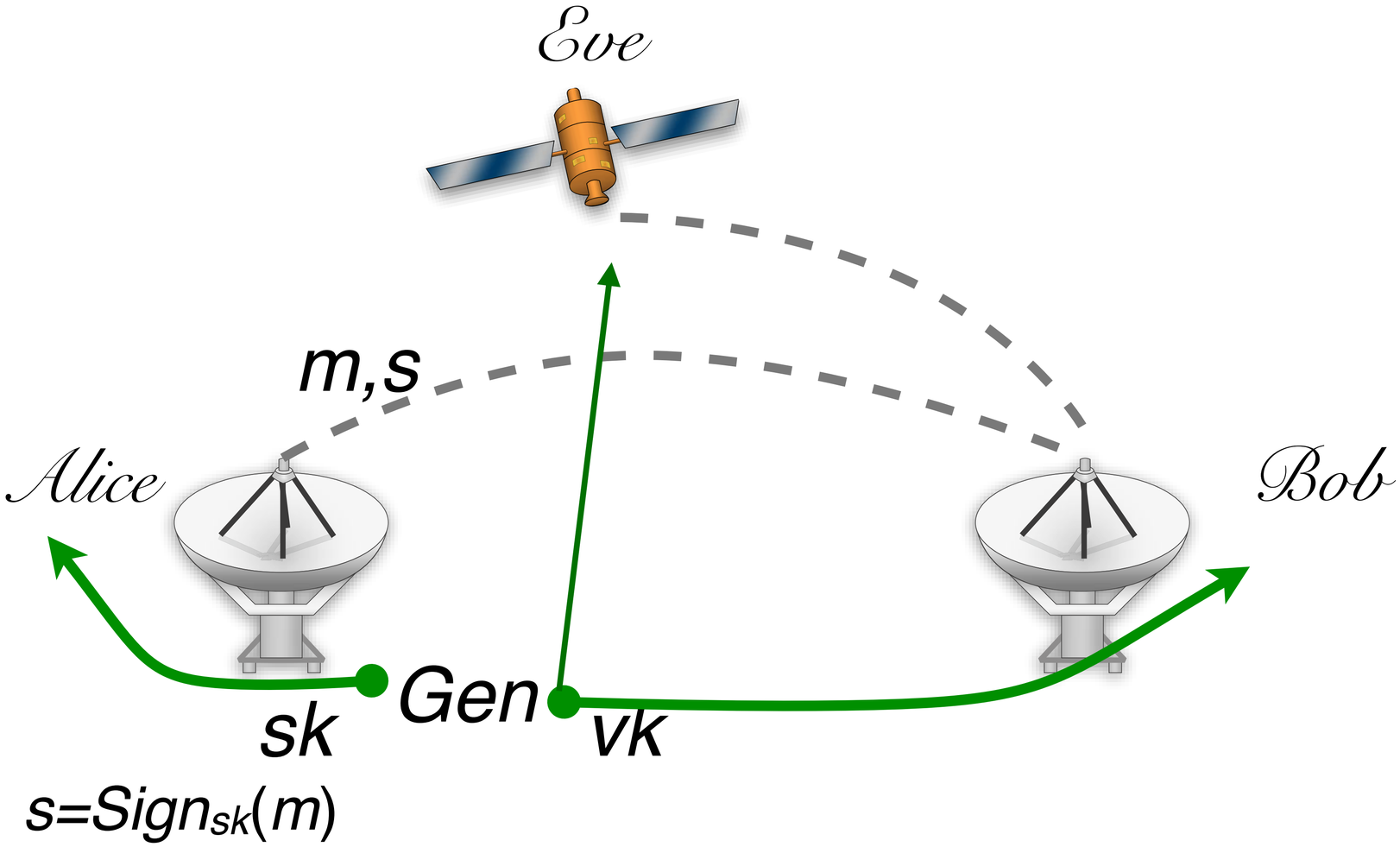
Public key digital signature



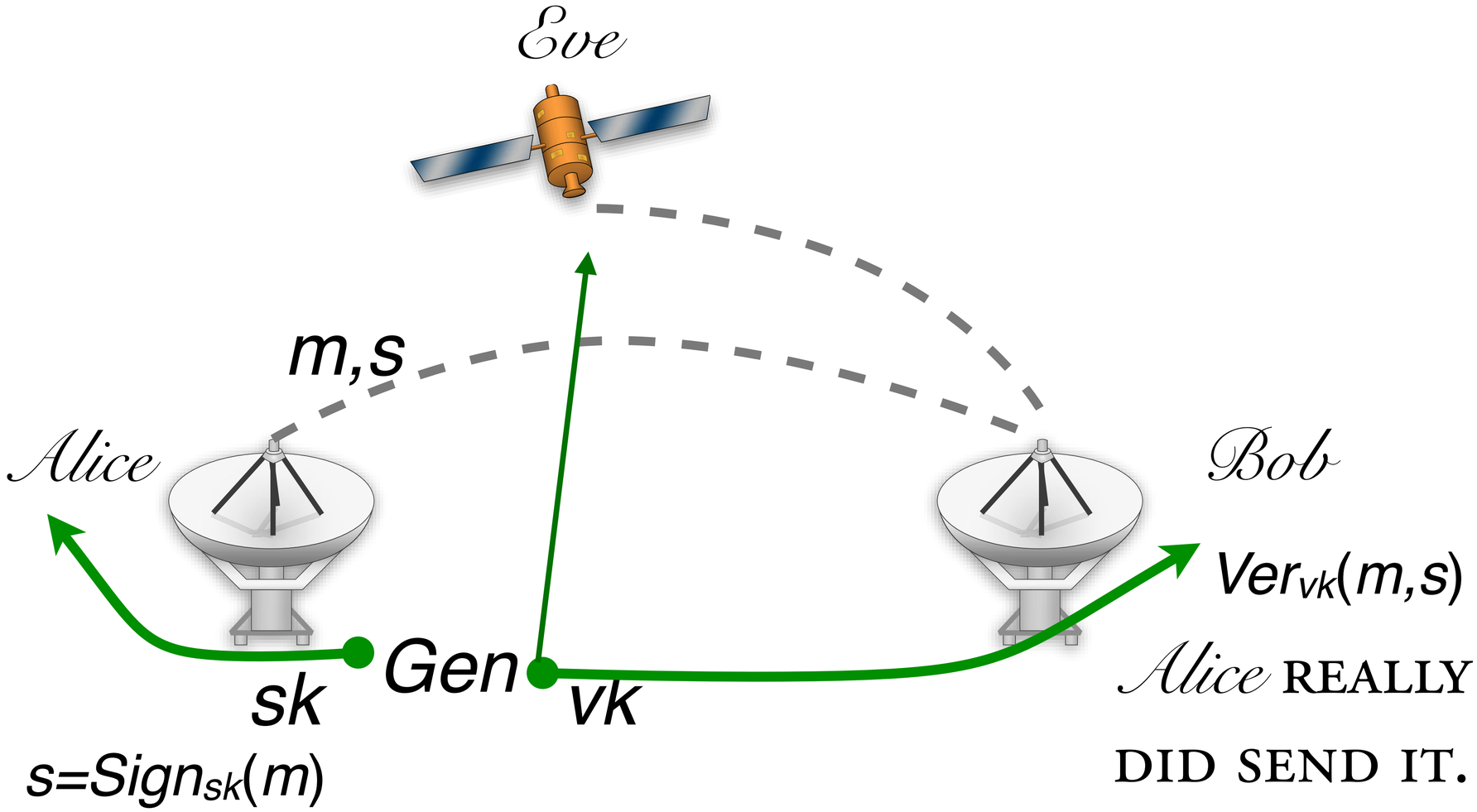
Public key digital signature



Public key digital signature



Public key digital signature



Public key digital signature

MESSAGE SPACE $\{\mathcal{M}\}_n$

$Gen(1^n)$

$Sign_{sk}(m)$

$Ver_{vk}(m,s)$

Public key digital signature

MESSAGE SPACE $\{\mathcal{M}\}_n$

$Gen(1^n)$ GENERATES A KEY PAIR sk, vk

$Sign_{sk}(m)$

$Ver_{vk}(m,s)$

Public key digital signature

MESSAGE SPACE $\{\mathcal{M}\}_n$

$Gen(1^n)$ GENERATES A KEY PAIR sk, vk

$Sign_{sk}(m)$ GENERATES A SIGNATURE s FOR $m \in \mathcal{M}_n$

$Ver_{vk}(m,s)$

Public key digital signature

MESSAGE SPACE $\{\mathcal{M}\}_n$

$Gen(1^n)$ GENERATES A KEY PAIR ***sk,vk***

$Sign_{sk}(m)$ GENERATES A SIGNATURE ***s*** FOR $m \in \mathcal{M}_n$

$Ver_{vk}(m,s)$ ACCEPTS OR REJECTS A MSG,SIG PAIR

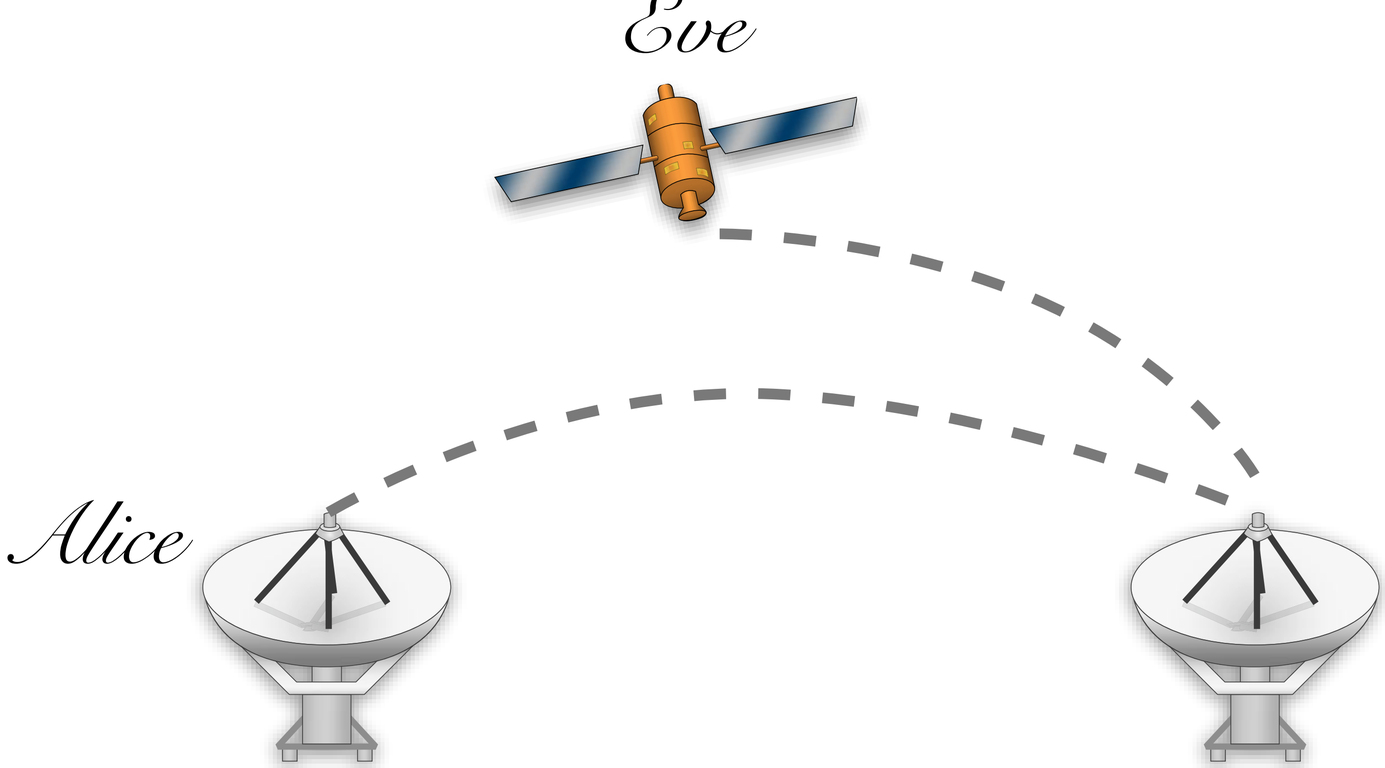
$$\Pr[k \leftarrow Gen(1^n) : Ver_{vk}(m, Sign_{sk}(m)) = 1] = 1$$

existential unforgability

"EVEN WHEN GIVEN A SIGNING ORACLE,

AN ADVERSARY CANNOT FORGE A SIGNATURE FOR

ANY MESSAGE OF ITS CHOOSING "

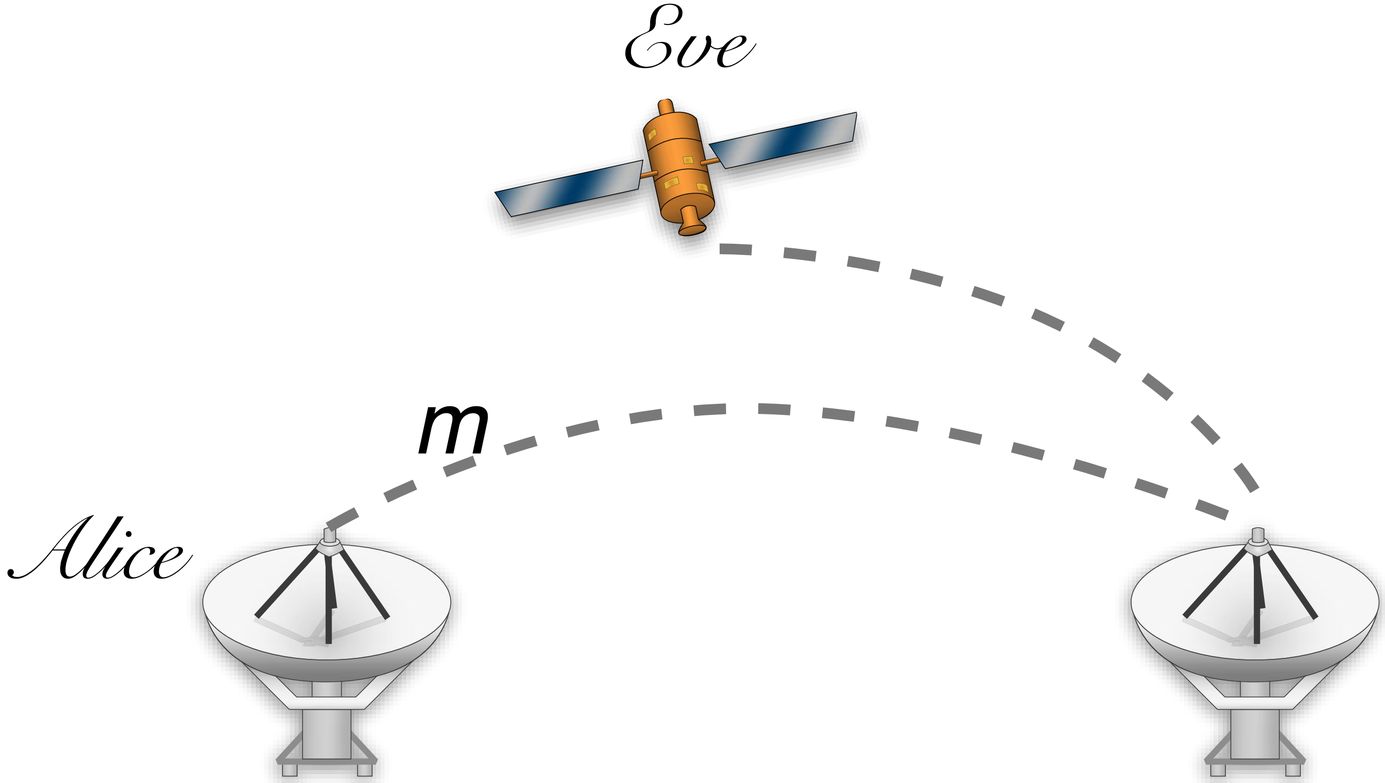


existential unforgability

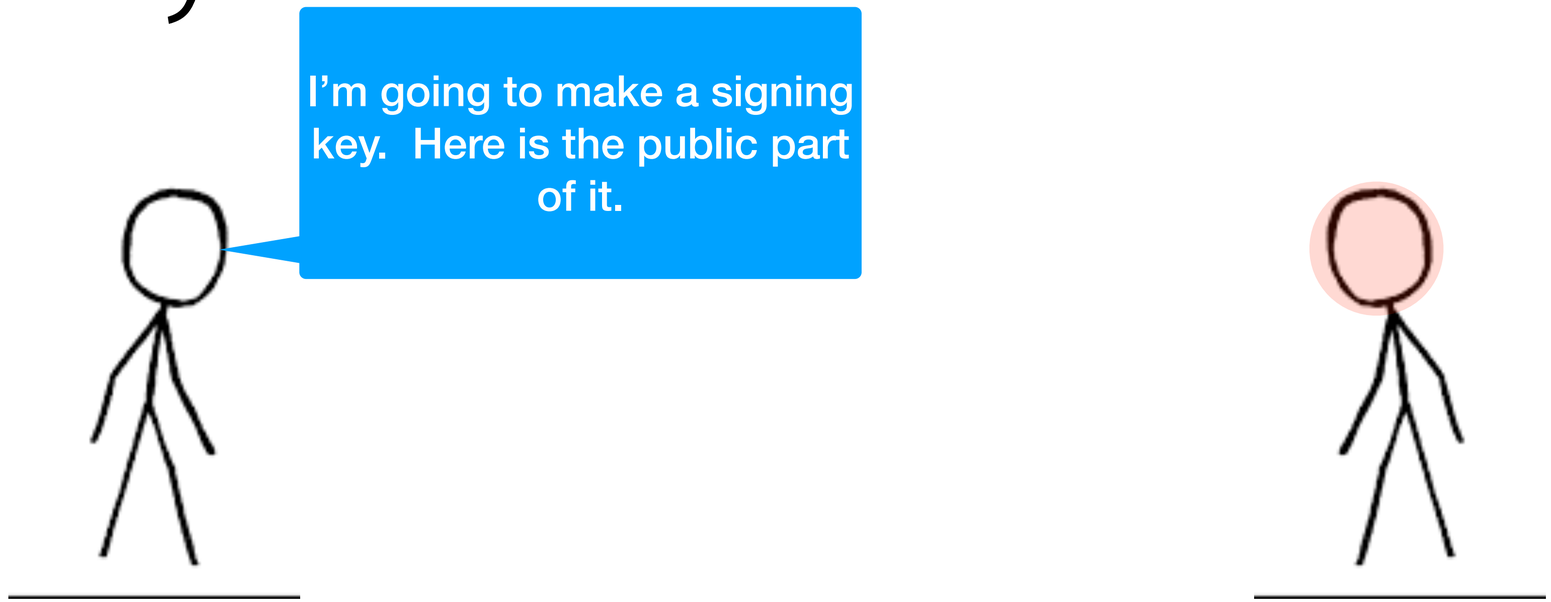
"EVEN WHEN GIVEN A SIGNING ORACLE,

AN ADVERSARY CANNOT FORGE A SIGNATURE FOR

ANY MESSAGE OF ITS CHOOSING "



Signature security



$(vk, sk) \leftarrow \mathrm{Gen}(1^n)$

Signature security



$(vk, sk) \leftarrow \mathrm{Gen}(1^n)$

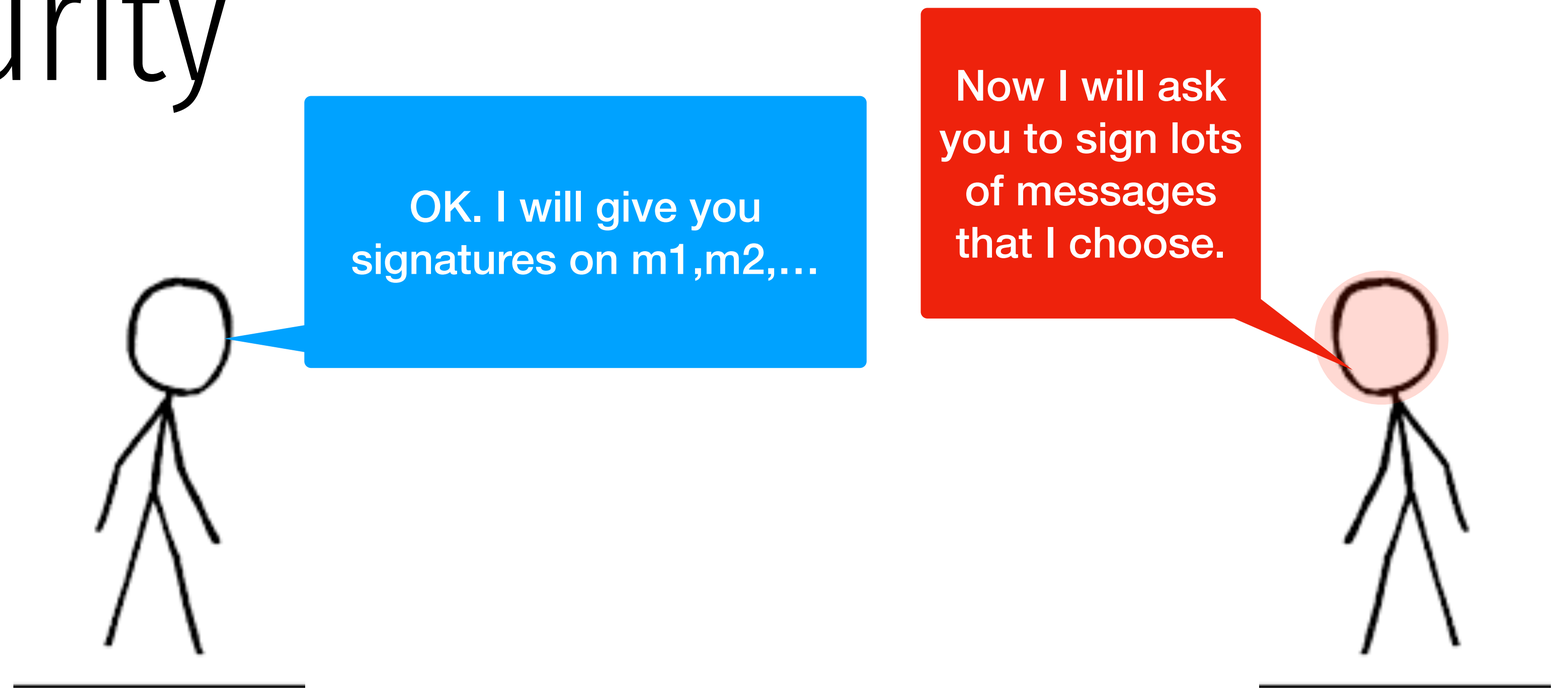
Now I will ask you to sign lots of messages that I choose.

$m_0, m_1, \ldots$



vk

Signature security



Signature security

Now I will try to create a new (signature, message) pair…one that I didn't receive from yoiu. signature on a new message





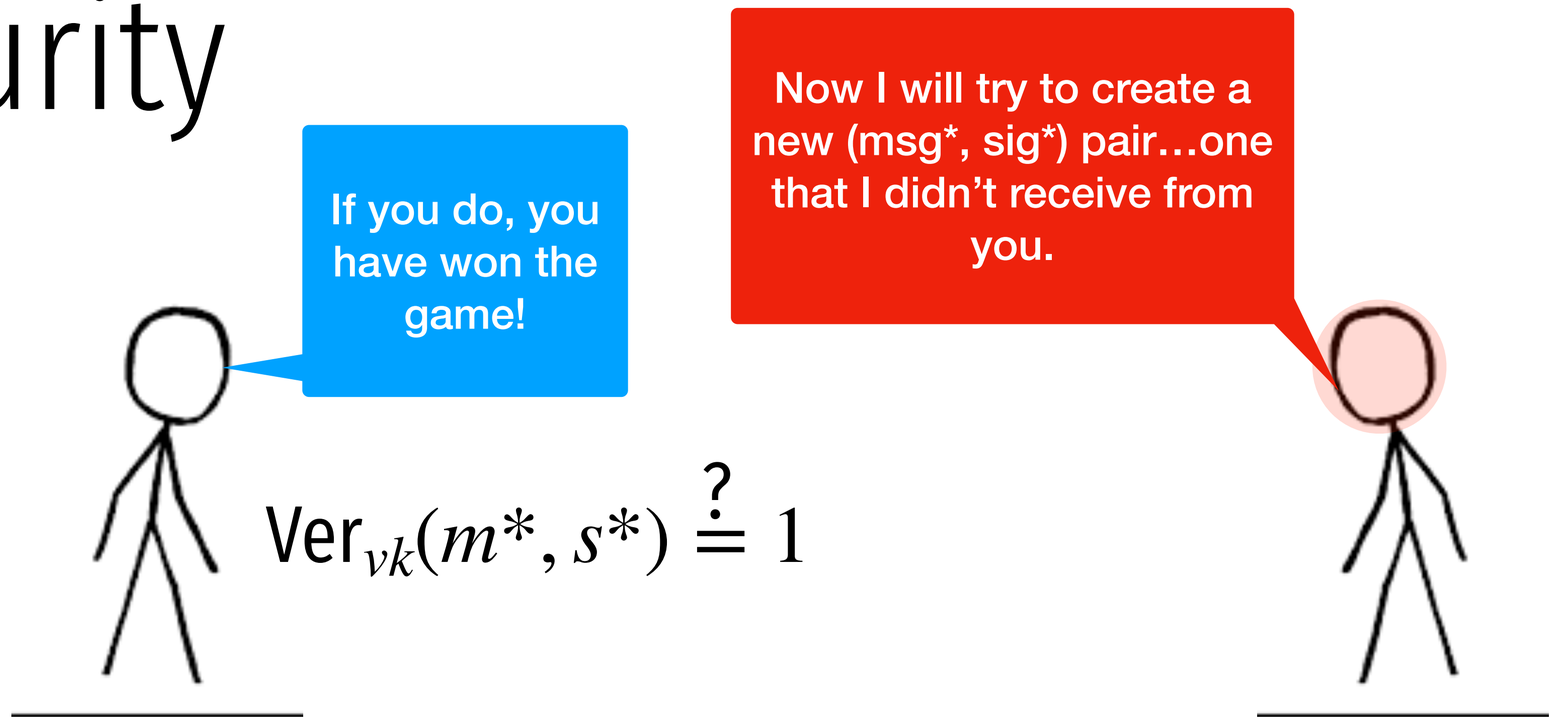
$(vk, sk) \leftarrow \text{Gen}(1^n)$

vk

$s_i \leftarrow \text{Sign}_{sk}(m_i)$

$s_1, s_2, \ldots$

Signature security



Textbook RSA Signatures (insecure)

Pick N = p*q where p,q are primes.

Pick e,d such that $e \cdot d = 1 \bmod \phi(N)$

Sign((sk=d, N) m):

Compute the signature: $\sigma \leftarrow m^d \bmod N$

Verify((pk=e, N), σ, m):

$$m \stackrel{?}{=} \sigma^e \bmod N$$

RSA Signatures in GPG

Sign((sk, N) m):

Compute the padding: $z \leftarrow 00 \cdot 01 \cdot FF \cdots FF \cdot 00 \cdot \mathsf{ID}_H \cdot H(m)$

Compute the signature: $\sigma \leftarrow z^{sk} \bmod N$