```
How long does the output of a CRHF home to be?
Birthday aftack on CRHFs. Suppose we have a hash function H: {0,15 > {0,13e How might we find a collision in 4 (without
                                  knowing anything more about H)
      Approach 1: Compute H(1), H(2), ..., H(2 + 1)
                                                                                                                                 Size of hash output space
                         L> By Pigeonhole Principle, there must be at least one collision — runs in time O(2^L)
      Approach 2: Sample m; & EONY and compute H(mi). Repeat until collision is found.
           How many samples needed to find a collision?
 Theorem (Birthday Paradox). Take any set S where |S| = n, Suppose r_1, \dots, r_d \stackrel{R}{\leftarrow} S. Then,
                                             \Pr\left\{\exists i \neq j : r_i = r_j\right\} \geqslant |-e^{-\frac{\ell(\ell-1)}{2n}}
Proof. P_{\Gamma}[\exists : f_{i} = f_{j}] = 1 - P_{\Gamma}[\forall i \neq j : f_{i} \neq f_{j}] conditioned on f_{i}, \dots, f_{i-1} being distinct
= 1 - P_{\Gamma}[f_{2} \notin \{f_{1}, f_{2}\}] \cdot P_{\Gamma}[f_{3} \notin \{f_{2}, f_{3}\}] \cdot \dots \cdot P_{\Gamma}[f_{Q} \notin \{f_{Q-1}, \dots, f_{Q}\}]
                                 = 1 - \frac{n-1}{n} \cdot \frac{n-2}{n} \cdot \dots \cdot \frac{n-l+1}{n}
                                                                                                                                  dominant term when |x| < 1
                                 = |-\prod_{i=1}^{\frac{n-1}{n}} \left(1 - \frac{i}{n}\right)  automatically holds for x \leq -1
                                 \geq |-\frac{1}{|1|} e^{-i/n} \quad \text{since } |+\chi \leq e^{x} \text{ for all } \chi \in \mathbb{R} \left[ e^{x} = |+\chi + \frac{x^{2}}{2} + \frac{\chi^{3}}{6} + \cdots \right]
                                 = |- e | - | = | - | = |
                                                                                                                                  positive for all x>0
                                               = 1 - e^{-\frac{(l-1)l}{2n}}
                                                                                         number of people in a room to have a common birthday
 When l ≥ 1.27 n, Pr[collision] = Pr[Ji+j: r=rj] > 1. [For birthdays, 1.2 √365 ≈ 23]
                                                                               Birthdays not aniformly distributed, but this only increases collision probability.
                                                                                                                                    [Try proving this!]
For hash functions with range 10/13^2, we can use a birthday attack to find collisions in time \sqrt{2^2} = 2^{1/2}
                                                                                                                                           can even do it with constant space!
     List For 128-bit security (e.g., 2128), we need the output to be 256-bits (hence SHA-256)
                                                                                                                                           via Floyd's cycle finding algorithm
     → Quantum collision-finding can be done in 243 (cube not attack), though requires more space
```

Back to building a secure MAC from a CRHF - can we do it more directly than using CRHF + small domain MAC?

Light Main difficulty seems to be that CRHFs are keyless but MACs are keyed

Toka: include the key as past of the hashed input

By itself, collision-resistance does not provide any "randomness" guarantees on the output

Light For instance, if H is collision-resistant, than H'(m) = moll...||mio|| H(m) is also collision-resistant even though H' also

leaks the first 10 bits/blocks of m

Constructing a PRF/MAC from a hash function will require more than just collision resistance

- Option 1: Model hash function as an "ideal hash function" that behaves like a fixed truly random function (modeling <u>Leuristic</u> called the random oracle model will encounter later in this course)
- Option 2: Start with a concrete construction of a CRHF (e.g., Merkle-Dampard or the sponge construction) and reason about its properties

L> We will take this approach

Several ways to build a layed function:  1. Prepared by: F(k, m) := H(k bar)  I Tescure due to otherwise of Mether Danghard: can maint an "extression obtack" given H(k    m), can compute H(k    m) by correctly Mether Danghard chain.  2. Approach by: F(k, m) := H(m    k)  I Souther to hard their MAC conservation. And whereable to some office estade: advancy finds a collision in the Mether Danghar point and was that to construct a fargery  In Structure copicital in SHP1 collision descendation (can general artifact; collision can probe reaches).  3. Everlope reached: F(k, m) := H(k    m    k)  I for reasonable precisionate structure con probe reaches).  4. The lay sery: F(kn, k), m) := H(k    m    k)  I for reasonable precisionate structure con probe reaches).  F(k, m) := A(k, m) = H(k, m    k), limit of reasonable precisionate structure can probe reaches).  HMAC is a PRF/MRC based on the tor-bay react (though with correlated laye):  HMAC is a PRF/MRC based on the tor-bay react (though with correlated laye):  HMAC is a PRF/MRC based on the tor-bay react (though with correlated laye):  HMAC is a PRF/MRC based on the tor-bay react (though with correlated laye):  HMAC is a PRF/MRC based on the tor-bay react (though with correlated laye):  HMAC is a PRF/MRC based on the tor-bay react (though with correlated laye):  HMAC is a PRF/MRC based on the tor-bay react (though with correlated laye):  HMAC is a PRF/MRC based to only a structure correlated and to south a structure of the structure of the south with the structure of the structure of the south with the structure of the structure of the south with the structure of the structure of the south with the structure of the structure of the south with the structure of the structure of the south with the structure of the structure of the south with the structure of the structure of the south with the structure of the structure of the south with the structure of the structure of the south with the structure of the south with the structure of the south with the structure of th	Suppose H	is a Merkle-Dan	ngard Lash Functio	n built from a	secure com	pression function		
1. Proposed key: F(k, m) := H(k, llm's)  L. Tenecoure due to structure of Mether Demograph: can mount an "entrasion attack" given H(k  m ), can compute H(k  m ) by considing Mether Demograph chain  2. Append key: F(k, m) := H(m  k)  L. Sindar to hood-than MAC construction, and inhereable to some offlice attack: advancy finds a collision in the Mather Demograph performance that to construct a forgety  L. Sindar to hood-than MAC construction and inhereable to some offlice attack: advancy finds a collision in the Mather Demograph performance to state the soft of the Legislation of the Machine Demograph (km) is a state of the Legislation of the Machine Demograph (km) in a state of the Legislation of the Machine to the State of								
Trescure due to structure of Merkle Dongland: can mount an "extression attack" given H(k   m), can compute H(k   m) by extending Merkle Pongland chain.  2. Approach bay: F(k,m):= H(m   k)  3. Employ to be settlen MRC construction and unlevable to some offline attack: advancy finds a collision in the Merkle Dongland probin and uses that to construct a forgery  1.5 Smaller Employe method: F(k,m):= H(k   m   k)  3. Emelope method: F(k,m):= H(k   m   k)  4. The lay Mat : F(k,k),m):= H(k   m   k)  5. The lay Mat : F(k,k),m):= H(k   m   k)  6. The lay Mat : F(k,k):= k,k); million of the construction one section TRFs on a social-size downship.  F(k,m):= K(k,m):= K(k,m):= K(k,m):= k,k); million of the construction one section TRFs on a social-size downship.  F(k,m):= H(k,   m   k); million by ant (though with correlation) by:  HMAC(k,m):= H(k,   m   k); million by ant (though with correlation) by:  HMAC(k,m):= K(m):= H(k,   m  k); million by and k; million construction one section TRFs on a social-size downship.  Collision and probable on the two bay ant (though with correlation) by:  HMAC(k,m):= K(m):= K(								
H(k  n  m') by extending Mether Posyerh chain  2. Approal bey: F(k,m) := f(m)  k)  -> Similar to hash than MAC exertination and violentials to some offline estable: advancy finds a collision in the Marker Despited in SHA-1 collision demonstration (an gracetic artistus; collisions once profite installed)  3. Eurologe method: F(k,m) = H(k  m  k)  1. The recognition of the fill   m  k   1. The recy nest: F((k, k, k, m) := H(k,   m  k)  1. The recognition of the fill   m  k   1. The recy nest: F((k, k, k, m) := H(k,   m  k)  1. The construction one security (k, m) and F(k, m) is a track-size domain  1. The construction one security (kg, h remains product and make a construction)  2. The recognition of the security of the standard of the construction one security (kg, h remains product and make a classification of the construction on security (kg, h remains product and make a classification of the security (kg, h remains product and make a classification of the security (kg, h remains product and make a classification of the security (kg, h remains product and make a classification of the security (kg, h remains product and make a classification of the security (kg, h remains product and make a classification of the security (kg, h remains product and make a classification of the security (kg, s decimal from a passessis)  1. The security is superior independent cryptographic keys from a single moster kg (kg, s decimal from a passessis)  1. The september independent cryptographic keys from a single moster kg (kg, s decimal from a passessis)  1. The september independent cryptographic keys from a single moster kg (kg, s decimal from a make kg from continuous passessis)  2. The september is maker and in TL3 and LTsec to derive seption (HKDF - RFC SB61)  2. Consists of two procedures:  2. Especial is define as interesting from the recovery seption of the procedures:  3. Especial is define as interesting from the recovery seption of the procedures:  3. Especial is define as interesting from the recovery seption of the pr				110 dd - D - 0 -		Nack-1	م الحال من من الحال من	
2. Append by: F(k,m):= H(n    k)  Solitar to bod-than MRC continuents and subscribt to some offline attack: advancy finds a collision in the Marke Dength and was that to construct a togety.  Merkle Danglish profit and was that to construct a togety.  Security of SNA-1, they such TDF the SNA-1 collision descentration (lan general artimax; collisions can profit mobiles).  3. Earninge related: F(k,m):= H(k,    H   k)  4. Ton-lay east: F(k,m):= H(k,    H   k),    H(k,    H   k)  4. Ton-lay east: F((k, k, k, m):= H(k,    H(k,    H   k))  And board MAPC  HMAC is a TRF/MAC board on the tour key rest (though with correlated lays):  HMAC(k, m):= H(k,    H(k, m))  Where k, — K B iped and k. — K B oped  and iped and oped are fined sproys (specifick in the HMAC standard)  DOS repeated  ONSC repeated  ONSC repeated  ONSC repeated  ONSC repeated  ONSC repeated  Totaletistus: Typically, deroined HMAC-H where H is the head function  e.g., HMAC-SHAI  HMAC-SHAI  HMAC-SHAIS — one of the recot windy-ward MAC on the cub Good in SSE/TIS, Tibec, SSH, and more)  HMAC for her related that under reasonable assumption, HMAC is a secure PRF  In many protects, see seed to derive multiple keys from a single master key (e.g., derived from a passonal)  Totaletistus  Totaletistus  Kence — HMAC (kousser, "com")  VRF 2000-1972 says derived keys one compositionally indivinguishable from kine — HMAC (kousser, "com")  VRF 2000-1972 says derived keys one compositionally indivinguishable from the passonal procedure:  Especial serve (e.g., a user passonal)  Especial derive substitute for muster by from our construction (HKDF — RFC SSE)  Especial derive substitute by from our construction for the muster by from our construction for the muster by from our construction of the procedures:  Especial derive substitute for muster by from our construction for the muster by from our construction of the						out extension	attack: guen H(K   m)	I, can compute
Horse Danghot profit and was that to Construct a forgery  → \$1911.1 Hay and \$15 ft.  Notice Danghot profit and was that to Construct a forgery  → \$1911.1 Hay and \$15 ft.  > Structure explained in \$14.1 ft.  \$2 Earnlope method: F(k, m):= H(k    m   k)  #4. Two-lay rest: F(k, m):= H(k    m   k)  #5. F(k, m):= h(k, m) and F(k, m) and F(k, m) to a rest; but the construction are secure \$18.5 on a variable-size domain than \$10.0 k, m):= H(k,    H(k, m))  #6. There constructions are secure \$18.5 on a variable-size domain than \$10.0 k, m):= H(k,    H(k, m))  #6. There is h(k, m):= H(k,    H(k, m))  #6. There is h(k, m):= H(k, m)  #6. There is h(k, m):= H(				rkle-langura cha				
Merkle Danglind profes and was that to construct a forgery  Structure explicital in SHPT collision demonstration (an against aristancy collisions once profits resolute)  3 Emologie method: F(k,m):= H(k,    m     k)  H. Tour key nest: F((k, k), m):= H(k,    H(k,    m))  Individual Security of the constructions assumptions on h (e.g., both film):= h(k,m) = h(k,m) is h(m,k) is a 18FF), both of these constructions are secure 18FD on a warable-size downship had beared MAC  HMAC is a PRF/MAC based on the tour key rest (though with correlated heps):  HMAC(k, m):= H(k,    H(k, m))  Dever k, & B iped and ke k B oped and ke k B oped and in the hadron of security:  Security: Since k, and ke are correlated, resed to make stronger assumption on security (e.g., h running pendormadum under a teletal-te				1 .	f1 /	fr	0 1 4.	
The standard in SHPT Collision derevolation (an governite artitlery collision once prefix matches)  3 Enrologe related is F(k,m) := H(k,    m  k)  4. Too lay rest: F((k, k_1),m) := H(k,    m  k)  5. F(k,m) := h(k,m) := h(k,m) is h(n,k) is a 1987), both of these constructions are secure 1985 on a surable-size domain the constructions are secure 1985 on a surable-size domain the constructions are secure 1985 on a surable-size domain the constructions are secure 1985 on a surable-size domain the constructions are secure 1985 on a surable-size domain of the k, m):  HMAC (k, m):= H(k,    H(k, m))  Where k, m k & piped and k, m k & poped and k, m k & poped and and iped and oped are fined springs (specified in the HMAC standard)  Octor repeated Octor repeated  Octor repeated Octor repeated  Security: Since k, and ks are correlated, need to make someter observation on security (e.g., h remains pseudorandom under a related to attending in HMAC shallows: Typically, devoted HMAC-H where H is the host function  e.g., HMAC-Shallows: Typically, devoted HMAC-H where Host of the need widely-used MAC on the coth (und in SSL/TLS, ITEC, SSH, and more)  HMAC shallows: Typically, devoted HMAC-H where most widely-used MAC on the coth (und in SSL/TLS, ITEC, SSH, and more)  HMAC-shallows: Recall that under researche assumption, HMAC is a secure PRF  In many protocols, we need to derive multiple keys from a single matter key (e.g., derived from a prosocod)  To derive multiple independent exployer-gaths keys, a PRF is a nestral printer:  kence — HMAC (knowley, "mac")  Respond to the procedures:  The operand paraligm is the express step in host-board key derived large are computationally individual bands in the most large.  Somet (e.g., a user passant)  — Extract define airthys from the most large.	-							
3. Employer method: F(km):= H(k 11 k)  H. Tourkey 1251: F((ki, ki, lm)):= H(kz    H(kz    m))  F(km):= h(km) and F(km):= h(n)k) is a TRE), but for these constructions are secure TRES on a survable-size domain. The form of these constructions are secure TRES on a survable-size domain. The form of the form of these constructions are secure TRES on a survable-size domain. The fill H(kz, m)  Where k, — k & poly and kz — k & popul and kz — k & popul and input and opul are fired shory; (specifich in the HMAC standard)  Disc repeated  OSSC repeated  OSSC repeated  Security: Since k, and kz are correlated, need to make stronger assumption on security (eg, h remains speculorandom under a related to the function and find the fired to the function of the fired to the fired to the function of the security in the fired to the function of the security in the fired to the function of the security in the fired to the function of the security in the fired to the function of the security in the fired to the function of the security in the fired to the function of the security in the fired to the function of the security in the fired to the function of the security in the fired to the function of the security in the fired to the function of the security in the fired the function of the security in the fired the function of the security in the fired function of the fired function of the security in the fired function of the security of								<b>\</b>
thost-board MAC  HMAC is a PRF/MAC based on the took key next (though with correlated keys):  HMAC(k, m) := H(k,    H(ks, m))  where k, ~ k & pipel and kz ~ k & oped  and iped and oped are fired strings (specified in the HMAC standard)  DUS repeated  OUSC repeated  OUSC repeated  Security: Since k, and ks are correlated, need to make stronger assumption on security (eg, h remains pecularized on with att  Institutions: Typically, durated HMAC H where H is the head function  eg, HMAC SHAI  HMAC SHAI  HMAC SHAI  HMAC share need to delive multiple keys from a single master key (eg, delived from a password)  To derive multiple independent cryptographic keys, a PRF is a master key (eg, delived from a password)  The derive multiple independent cryptographic keys, a PRF is a master key (eg, delived from a password)  This approach is seed in TLS and IPsec to derive seesing keys durin seesing setup  This approach is seed in TLS and IPsec to derive seesing keys durin seesing setup  This approach is seed in TLS and IPsec to derive seesing keys durin seesing setup  The General paradigm is the "expand" step in head-board key during seesing setup  The General paradigm is the "expand" step in head-board key during seesing the procedures:								
thost-board MAC  HMAC is a PRF/MAC based on the took key next (though with correlated keys):  HMAC(k, m) := H(k,    H(ks, m))  where k, ~ k & pipel and kz ~ k & oped  and iped and oped are fired strings (specified in the HMAC standard)  DUS repeated  OUSC repeated  OUSC repeated  Security: Since k, and ks are correlated, need to make stronger assumption on security (eg, h remains pecularized on with att  Institutions: Typically, durated HMAC H where H is the head function  eg, HMAC SHAI  HMAC SHAI  HMAC SHAI  HMAC share need to delive multiple keys from a single master key (eg, delived from a password)  To derive multiple independent cryptographic keys, a PRF is a master key (eg, delived from a password)  The derive multiple independent cryptographic keys, a PRF is a master key (eg, delived from a password)  This approach is seed in TLS and IPsec to derive seesing keys durin seesing setup  This approach is seed in TLS and IPsec to derive seesing keys durin seesing setup  This approach is seed in TLS and IPsec to derive seesing keys durin seesing setup  The General paradigm is the "expand" step in head-board key during seesing setup  The General paradigm is the "expand" step in head-board key during seesing the procedures:	3 E	welope method:	F(k,m) := H(k) r	1    k)	for reason	able pseudorandom	mess assumptions on	h (e.g., both
HMAC is a PRF/MAC based on the tour lay next (though with correlated keys):  HMAC(k, m):= H(k,    H(ks, m))  where k, ~ k & piped and ke ~ k & opped  and iped and oped are fired surjey (specified in the HMAC standard)  Oxis repeated OxisC repeated  Security: Since k, and ke are correlated, need to make stronger assumption on security (e.g., h remains pseudorandom under a related to att Institutions: Tyrically, devoted HMAC-H where H is the host function  e.g., HMAC-SHA256 — one of the much widely-used MAC on the curb (used in SSL/TCS, IBec, SSH, and more)  HMAC for hy-deviation: Recall that under reasonable assumptions, HMAC is a secure PRF  In many protects, we need to derive multiple keys from a single master key (e.g., derived from a password)  > To derive multiple independent cryptographic keys, a PRF is a matural printine:  kenc — HMAC (knowner, "cnc")  Assured keys master key tray (just has to be unique)  This approach is used in TL3 and IPsec to derive section keys durin session flittle—RFC 5884)  — General parallym is the "expand" step in had board key derivation (HKCF—RFC 5884)  — Consists of two procedures:  - Extract: derive a master key from the master key  Source (e.g., a user password)  - Expand: derive as subtracy from the master key	4. Tu	o-key nest: $F(($	.k1, k2), m) := H(k	2    H(k,    m))				
HMAC is a PRF/MAC based on the train key act (though with correlated keys):  HMAC(k, m) := H(k,    H(ks, m))  where k, — k @ ipad and kz — k @ opad  and ipad and opad are fixed surges (Specified in the HMAC standard)  Octor repeated Ox50 repeated  Security: Since k, and kz are correlated, reed to make stronger assumption on security (eg., h remains speculorandom under a related to att  Institutions: Typically, denoted HMAC-H where H is the heat function  eg., HMAC 5HA1  HMAC 5HA256 — one of the most widely-used MAC on the web (used in SSL/TCS, ITBec, 55H, and more)  HMAC for bey-decisation: Recall that under reasonable assumptions, HMAC is a secure PRF  In many protocols, we need to derive multiple keys from a single matter key (eg., obvioed from a password)  > To derive multiple independent cryptographic keys, a PRF is a natural primitive:    kenc	had hased	MAC.			of these o	constructions are se	cure PRFs on a varial	de-size domain
HMAC(k, m) := H(k,    H(ks, m))  where k, ← k ⊕ ipod and kz ← k ⊕ opod  and ipod and opod are fixed strongs (specified in the HMAC standard)  DXS expected OxSC repeated	1/000K- Breakty	, , , , , ,						
Security: Since h, and ke are correlated stronger operated in the HMAC standard)  Dick repeated Ox5C repeated  Security: Since h, and ke are correlated, need to move stronger assumption on security (e.g., h remains seculorandom under a related to at Institutions: Typically, devoted HMAC-H where H is the heat function  e.g. HMAC-SHAZ56— one of the most widely-used MAC on the cub (used in SSL/TCS, IBEC, SSH, and move)  HMAC for key-devication: Recall that under reasonable assumptions, HMAC is a secure PRF  In many protects, use need to derive multiple keys from a single moster key (e.g., deviced from a password)  To derive multiple independent cryptographic keys, a PRF is a natural printine:  kenc — HMAC (knowner, "cnc")  JRF security says derived begins are computationally indistinguishable from know — HMAC (knowner, "cnc")  This approach is used in TL3 and IPsec to derive session keys durin session setup  General paradigm is the "expand" step in heat-board key derivation (HKDF— RFC 5861)  — General paradigm is the "expand" step in heat-board key derivation (HKDF— RFC 5861)  — Extract: darive substitute from the mass source (e.g., a user passand)  — Extract: darive substitute from the mass	HMAC is a	PRF/MAC base	ed on the two-key	y nest (though u	with correlated	keys):		
where k, \( \sim \text{ k B pold and kz} \sim \text{ k B oped} \)  and ipad and oped are fired strings (specified in the HMAC standard)  Dick expected \( \text{Ox5C repeated} \)  Security: Since k, and kz are correlated, need to make stronger assumption on security (e.g., h remains pseudorandom under a related to at \)  Tastentiations: Typically, deroted. HMAC-H where H is the head function  e.g., HMAC-SHA256 — one of the most widely-used MAC on the web (used in SSL/TCS, ITBEC, SSH, and more)  HMAC for key-derivation: Recall that under reasonable assumptions, HMAC is a secure PRF  In many protects, we need to derive multiple keys, a PRF is a natural printitre:  kenc \( \text{HMAC (knowner, "enc")} \)  Reac \( \text{HMAC (knowner, "enc")} \)  VRF security says derived keys are computationally indistinguishable from kines \( \text{HMAC (knowner, "enc")} \)  This approach is used in TC3 and ITsec to derive session keys durin session setup  General paradigm is the "expand" step in host-based key derivation (HKDF - RFC SBEN)			,					
and ipod and oped are fired strings (specified in the HMAC standard)  Octo repeated  Octo repeat				k2 ← k @ opa	l l			
Security: Since k, and ke are correlated, need to make stronger assumption on security (eg., h remains specularizedom under a related to att  Tasineticions: Typically, denoted HMACH where H is the hook function  eg., HMAC-SHA1  HMAC-SHA256 — one of the most widely-used MAC on the under (used in SSC/TLS, IRsec, SSH, and more)  HMAC-SHA256 — one of the most widely-used MAC on the under (used in SSC/TLS, IRsec, SSH, and more)  HMAC-SHA256 — one of the most widely-used MAC on the under (used in SSC/TLS, IRsec, SSH, and more)  HMAC-SHA256 — one of the most widely-used MAC on the under (used in SSC/TLS, IRsec, SSH, and more)  HMAC-SHA256 — one of the most widely-used MAC on the under (used in SSC/TLS, IRsec, SSH, and more)  HMAC-SHA256 — one of the most widely-used MAC on the under						C Standard)		
Society: Since h, and ke are correlated, need to make stronger assumption on security (eg., h remains special random under a related to att  Institution: Typically, denoted HMACH where H is the host function  eg., HMAC-SHA1  HMAC-SHA2E6— one of the most widely-used MAC on the web (weed in SSL/TLS, IRec, SSH, and more)  HMAC for key-derivation: Recall that under reasonable assumptions, HMAC is a secure PRF  In many protects, we need to derive multiple keys from a stude master key (eg., derived from a passaonal)  To derive multiple independent cryptographic keys, a PRF is a natural primitive:    Kenc HMAC (knowless, "enc")   PRF security says derived keys are competitionally indistinguishable from knowless have master key  This approach is used in TLS and IRec to derive session keys durin session setup  General paradigm is the "expand" step in hash-based key derivation (HKDF—RFC 5069)    Consists of two procedures:		7	Č	0 (				
Security: Since h, and kx are correlated, need to make stronger assumption on security (eg., h remains specularandom under a related to att  Institution: Typically, denoted HMACH where H is the heat function eg., HMAC-SHA1  HMAC-SHA256 — one of the most winder-used MAC on the undo (used in SSL/TLS, ITSEC, SSH, and more)  HMAC for key-decision: Recall that under reasonable assumptions, HMAC is a secure PRF  In many protocols, are need to durine multiple keys from a single master key (eg., decised from a password)  > To derive multiple independent cryptographic keys, a PRF is a natural prinitive:    kenc — HMAC (kmaster, "enc")   PRF security says derived keys are computationally indistinguishable from		0x36 repeated	0x5C repeate	d				
Instructions: Typically, denoted HMAC-H where H is the host function  e.g., HMAC-SHA1  HMAC-SHA256 — one of the most widely-used MAC on the useb (used in SSL/TLS, ITBEC, SSH, and more)  HMAC for key-defination: Recall that under reasonable assumptions, HMAC is a secure PRF  In many protects, we need to derive multiple keys from a simple master key (e.g., defined from a password)  To derive multiple independent cryptographic keys, a PRF is a natural prinitive:  kenc — HMAC (knowner, "enc")  PRF security sous derived keys are computationally indistinguishable from k.mac — HMAC (knowner, "enc")  derived keys  master key  trog (just has to be unique)  This approach is used in TLS and IPsec to derive session keys durin session setup  General paraligm is the "expand" step in hash-based key derivation (HKDF - RFC 5861)  Consists of two procedures:  Estract: durine a master key from the master key keys from the master keys  Expand derive sub-keys from the master keys  Expand derives sub-keys from the master keys  Expand derives sub-keys from the master keys	c 0					/		
Instructions: Typically, denoted HMAC-H where H is the host function  e.g., HMAC-SHA1  HMAC-SHA256 — one of the most widely-used MAC on the useb (used in SSL/TLS, ITBEC, SSH, and more)  HMAC for key-defination: Recall that under transmoble assumptions, HMAC is a secure PRF  In many protects, we need to derive multiple keys from a single master key (e.g., defined from a password)  To derive multiple independent cryptographic keys, a PRF is a natural prinitive:  kenc — HMAC (knowner, "enc")  PRF security soys derived keys are computationally indistinguishable from k.mac — HMAC (knowner, "enc")  derived keys are computationally indistinguishable from the approach is used in TLS and IPsec to derive session keys durin session setup  Generally parallym is the "expand" step in host-based keys durination (HKDF - RFC 5861)  Extract: darive a master key from the mast source (e.g., a user password)  Excand derive sub-keys from the mast key	Security: Since	e k, and ke an	e correlated, need	to make stronge	ir assumption o	n security (e.g., h	remains pseudorandom w	ider a <u>related-la</u>
eg, HMAC-SHA256 — one of the most widely-used MAC on the useb (used in SSL/TLS, IRSEC, SSH, and more)  HMAC for bey-decisedion: Recall that under reasonable assumptions, HMAC is a secure PRF  In many protects, use need to derive multiple keys from a single master key (eg., obsined from a passaund)  > To derive multiple independent cryptographic keys, a PRF is a natural primitive:    Kenc = HMAC (kmaster, "enc")   PRF security says derived keys are computationally indistinguishable from kmac = HMAC (kmaster, "enc")   uniform								
HMAC for key-decision: Recall that under reasonable assumptions, HMAC is a secure PRF  In many protects, we need to derive multiple keys from a single moster key (e.g., oberived from a password)  To derive multiple independent cryptographic keys, a PRF is a notural primitive:    Kenc — HMAC (kmaster, "enc")   PRF security says derived keys are computationally indistinguishable from kmac — HMAC (kmaster, "mac")   uniform	<u>Instantiations</u> :	Typically, denoted	HMAC-H where	H is the hash	<i>tunction</i>			
HMAC for key-decisation: Recall that under reasonable assumptions, HMAC is a secure PRF  In many protocols, we need to derive multiple keys from a single master key (e.g., decived from a password)  > To derive multiple independent cryptographic keys, a PRF is a notural primitive:    Kenc — HMAC (knoster, "enc")   PRF security says derived keys are computationally indistinguishable from known — HMAC (knoster, "mac")   uniform — defined keys moster key tog (just has to be unique)  This approach is used in TLS and IPsec to derive session keys durin session setup   > General paradigm is the "expand" step in hash-based key derivation (HKDF - RFC 5869)   > Consists of two procedures:	· .							
In many protocols, we need to derive multiple keys from a single master key (e.g., obsided from a passassed)  To derive multiple independent cryptographic keys, a PRF is a notural primitive:  Kenc — HMAC (kmaster, "enc")  PRF security says derived keys are computationally indistinguishable from kmac — HMAC (kmaster, "mac")  duried keys  master key tag (just has to be unique)  This approach is used in TLS and IPsec to derive session keys durin session setup  —> General paradigm is the "expand" step in hash-based key-derivation (HKDF - RFC 5869)  —> Consists of two procedures:  Extract: derive a master key from ent source (e.g., a user passassed)  — Expand: derive subtrees from the master	ЭАМН	-SHA256 — one	of the most wid	ely-used MAC or	the web (us	ed in SSL/TLS, IPse	ec, SSH, and more)	
In many protocols, we need to derive multiple keys from a single master key (e.g., durined from a password)  To derive multiple independent cryptographic keys, a PRF is a natural primitive:  Kenc   HMAC (kmaster, "enc")  PRF security says derived keys are computationally indistinguishable from kmac   HMAC (kmaster, "mac")  durined keys  master key  tag (just has to be unique)  This approach is used in TLS and IPsec to derive session keys durin session setup  General paradigm is the "expand" step in hash-based key-derivation (HKDF-RFC 5869)  Consists of two procedures:  Extract: derive a master key from enther source (e.g., a usex password)  Expand: derive subtrays from the master key				•				
Les to derive multiple independent cryptographic keys, a PRF is a natural primitive:    Kenc HMAC (Kmaster, "enc")   PRF security says derived keys are computationally indistinguishable from kmac HMAC (Kmaster, "mac")   uniform	HMAC for he	y-decination: Rea	all that under reason	onable assumptions	, HMAC is a	L secure PRF		
Les to derive multiple independent cryptographic keys, a PRF is a natural primitive:    Kenc HMAC (Kmaster, "enc")   PRF security says derived keys are computationally indistinguishable from kmac HMAC (kmaster, "mac")   uniform	In many	protocols, we nee	d to derive multip	le keys from	a single maste	er key (e.g., derive	d from a password)	
kenc   HMAC (kmaster, "enc")  PRF security says derived keys are computationally indistinguishable from kmac   HMAC (kmaster, "mac")  derived keys moster key tag (just has to be unique)  This approach is used in TLS and IPsec to derive session keys durin session setup  Seneral paradigm is the "expand" step in hash-based key derivation (HKDF - RFC 5869)  Consists of two procedures:  Extract: derive a master key from ent source (e.g., a user password)  Expand: derive sub-keys from the master key								
k mac   HMAC (kmaster, "mac")  Just has to be unique)  This approach is used in TLS and IPsec to derive session tays durin session setup  -> General paradigm is the "expand" step in hash-based key-derivation (HKDF - RFC 5869)  -> Consists of two procedures:  Extract: darine a master key from entropy source (e.g., a user password)  Expand: derive sub-keys from the mast								tinguishable from
derived keys master key tog (just has to be unique)  This approach is used in TLS and IPsec to derive session keys durin session setup  Seneral paradigm is the expand" step in hash-based key-derivation (HKDF - RFC 5869)  Consists of two procedures:  - Extract: derive a master key from entering source (e.g., a user password)  - Expand: derive sub-keys from the master key							\ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \	0
This approach is used in TL3 and IPsec to derive session keys durin session setup  -> General paradigm is the "expand" step in hash-based key-derivation (HKDF - RFC 5869)  -> Consists of two procedures:  Extract: derive a master key from ent  50urce (e.g., a user password)  Expand: derive sub-keys from the most								
This approach is used in TL3 and IPsec to derive session keys durin session setup  -> General paradigm is the "expand" step in hash-based key-derivation (HKDF - RFC 5869)  -> Consists of two procedures:  Extract: derive a master key from ent  50urce (e.g., a user password)  Expand: derive sub-keys from the most		derived keys	master key	tag (just has	to be unique)			
General paradigm is the "expand" step in hash-based key-derivation (HKDF - RFC 5869)  Consists of two procedures:  - Extract: derive a master key from entermined to source (e.g., a user password)  - Expand: derive sub-keys from the master key	<b>T</b> .							
Consists of two procedures:  - Extract: derive a master lay from enter  50urce (e.g., a user password)  - Expand: derive sub-keys from the mast							500	
- Extract: derive a master buy from ent  50urce (e.g., a user password)  - Expand: derive sub-keys from the mast	<b>□</b> (	several paradigm	is the expand	step in hash-bas	ed key-derivat			
50urce (e.g., a user possword)  - Expand: derive sub-keys from the mast						- Consists	of two procedures:	
Expand: derive sub-keys from the most						<u> </u>	itract: derive a maste	ckey from ent
Expand: derive sub-keys from the most							source leig,, a c	user powerd)
Lay (						<u> </u>		1 1 1
Both Steps rely on HMAC							lay (	
						Both sto	ps rely on HMAC	