EXHIBIT C

		Page 1
IN THE UNITED STATES	DISTRICT COURT	
FOR THE EASTERN DIST	RICT OF TEXAS	
TYLER DIVISIO	N	
* * * * * * * * * *	* * * * * * * *	
BEDROCK COMPUTER TECHNOLOGIES, LLC,	Civil Action No.	
Plaintiff	6:09-CV-269-LED	
VS.		
SOFTLAYER TECHNOLOGIES, INC., ET AL.,		
Defendants		
* * * * * * * * * *	* * * * * * * *	
VIDEOTAPED DEPOSITION OF:	ALEXEY KUZNETSOV	

Taken before Isabelle Klebanow, Registered Professional Reporter and Notary Public, pursuant to Notice, at the offices of Goltsblat BLP Law Firm, Capital City Complex, Moscow City Business Center, 8,

Prennenskaya Nab. Building 1, Moscow, Russia, on

Thursday, January 27, 2011, commencing at 1:15 p.m.

	Page 38		Page 40
1	A. Yes. Exactly.	1	function. Then it goes to the original code.
2	Q. Okay. And did you testify that the lines below	2	Q. Okay. Right below the comment that says Cleanup
3	this comment remove records?	3	duplicate and aged off entries, I think one, two
4	A. Yes.	4	three lines below it, there's a line that starts with
5	Q. And did you also testify that, going back toward	5	the word While.
6	the top of the page, the rt_garbage_collect function	6	Do you see that?
7	also removed records?	7	A. Yes.
8	A. Yes.	8	Q. Do you know what the purpose of that line is?
9	Q. Okay. So is it your understanding that there are	9	A. Yes. This line start of cycle which counts
10	two portions of code in rt cache-add that remove	10	through hash table chain.
11	records?	11	Q. And do you see, immediately below the while
12	A. Yes.	12	statement, do you see an opening brace?
13	Q. What was the purpose strike that.	13	A. Yes.
14	Why did you put two different places strike	14	Q. What does that signify?
15	that. I'm sorry. Let me try to ask the question more	15	A. That everything which after this line until the
16	correctly.	16	closing brace is used is executed for each entry in
17	Why did you include two sections of code in this	17	this hash table chain.
18	function that each remove records?	18	Q. And where is the closing brace that you referred
19	A. Even though they do the same thing, they work	19	to?
20	quite differently.	20	A. Closing brace is on the next page, on the first
21	Rt the first way, which does globals, can't	21	line on the next page.
22	remove entry, is much more efficient in the sense that	22	Q. Okay. And so, just to be clear, is the closing
23	it counts all the tables. So you can't write to the	23	brace you're referring to on page 9314, and it's
24	situation when one part of table is more empty and	24	immediately above the restore_flags line?
25	another has a lot of entries. So when we add an entry	25	A. Yes.
	Page 39		Page 41
1	to some part of hash table, this counts all the table to	1	Q. So is it your testimony that, between the opening
2	balance through all of it.	2	brace under the while loop and that closing brace at the
3	From the other hand, it is a slow process. We	3	top of 9314, that the code in between those two braces
4	have to count through all the table. It's expensive.	4	is executed in the while loop?
5	It's much cheaper to remove some entries which are	5	A. Yes.
6	already under hand. We add them to some hash line.	6	Q. Okay. Going back to the rt_garbage_collect
7	We already can count forward through the list to	7	function, did you testify that that function is a
8	find some entries to remove right there, in hope then,	8	garbage-collection process?
9	if we are lucky, hash table size will not grow above the	9	A. It's just a term. I I the term wasn't
10	limit and the first function, rt_garbage_collect, will	10	written by me. But probably it's just a term used by
11	probably even never will be triggered.	11	software programmers for garbage collection is a
12	Q. You referred to the rt_garbage_collect function	12	process of removal all unneeded data, and they link the
13	as expensive, is that correct?	13	sources to the system.
14	A. Yes.	14	So, yes, I testified that the function
15	Q. What do you mean by expensive?	15	rt_garbage_collect do garbage collection.
16	A. It takes quite a lot CPU resources, CPU time, to	16	Q. And so is do you know if the code within the
1 7		117	vylada la angla angla manfannang 1 11 0

17

18

19

20

21

22

A. It is.

this function?

A. Yes.

- 10 At a traces quite a lot of o resources, of o time, to
 17 count through all the hash table, which could be quite
 18 large in size.
 19 Q. Okay. And just to back up just one more time. I
- think you testified that rt_cache-add callsrt_garbage_collect?
- 21 rt_garbage_co22 A. Yes.
- Q. What does it mean to call rt_garbage_collect?
- A. To use this function as part of work, and then it
- calls up -- it uses some piece of code from the
- 23 identification.)
 24 Q. Okay. I am handing you what has been marked as
 25 Exhibit 3 -- not just yet.

(Exhibit No. 3 marked for

while loop is also performing garbage collection?

Q. So you have two garbage-collection routines in

	Page 42		Page 44
1	I am handing you what has been marked as	1	Q. Okay. And so those numbers on the bottom of the
2	Exhibit 3 (indicating). If you go by the Bates numbers	2	page correspond to the numbers in paragraph 6?
3	at the bottom right side of the page, it begins with	3	A. Yes.
4	DEF00001013 and it ends on DEF00001043.	4	Q. Okay. So and when you signed the declaration
5	A. Yes.	5	on December 15, 2010, you personally verified that this
6	Q. Do you recognize this document?	6	was that the version of Linux 1.3.52
7	A. Yes. This is a doc content of the file. It is	7	A. Yes.
8	that file. It is a file of rt.	8	Q represented by these pages is a true and
9	Q. Okay. And who who wrote this file?	9	correct copy?
10	A. It was written by many people. But, actually, I	10	A. Yes.
11	think 90 percent or more of this file was written by me.	11	Q. Okay. Let's turn to page let's see to
12	Q. And when did you write strike that.	12	page 1036. And just let me know when you find that
13	So do you see your name on the first page of this	13	page.
14	document?	14	A. Okay. I found it.
15	A. Yes.	15	Q. Okay. Thank you. Do you see a line around the
16	Q. So were you one of the people who wrote this	16	middle of the page that starts with, Static void
17	document?	17	rt_cache-add?
18	A. Yes.	18	A. Yes.
19	Q. Okay. And when did you and so you wrote a	19	Q. Okay. And is that the same function rt_cache-add
20	portion of this document, is that right?	20	that we discussed earlier today?
21	A. Yes.	21	A. Yes.
22	Q. And other people wrote other portions?	22	Q. Okay. And do you know if any other portions of
23	A. Yes.	23	route.c use the rt_cache-add function?
24	Q. When did you write your portion of this document?	24	A. Yes.
25	A. I wrote it in 1995.	25	Q. Do you know what portions those are?
	Page 43		
	raye 45		Page 45
1	Q. Okay. And is the portion that you wrote the	1	A. I can find it. A function for example, it's a
1 2	Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with	1 2	A. I can find it. A function for example, it's a function next to rt_cache-add function,
	Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with Exhibit 2, which is the patch?		A. I can find it. A function for example, it's a function next to rt_cache-add function, ip_rt_slow_route. The function is responsible for
2	Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with Exhibit 2, which is the patch? A. Yes.	2 3 4	A. I can find it. A function for example, it's a function next to rt_cache-add function, ip_rt_slow_route. The function is responsible for creation of a new routing cache entry.
2 3 4 5	Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with Exhibit 2, which is the patch?A. Yes.Q. Okay. So take a look at this, and I want to ask	2 3 4 5	A. I can find it. A function for example, it's a function next to rt_cache-add function, ip_rt_slow_route. The function is responsible for creation of a new routing cache entry. And then close to end it's page 1039 it
2 3 4 5 6	Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with Exhibit 2, which is the patch?A. Yes.Q. Okay. So take a look at this, and I want to ask you if this is a true and correct copy of route.c from	2 3 4 5 6	A. I can find it. A function for example, it's a function next to rt_cache-add function, ip_rt_slow_route. The function is responsible for creation of a new routing cache entry. And then close to end it's page 1039 it calls rt_cache-add.
2 3 4 5 6 7	Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with Exhibit 2, which is the patch?A. Yes.Q. Okay. So take a look at this, and I want to ask you if this is a true and correct copy of route.c from Linux version 1.3.52.	2 3 4 5 6 7	 A. I can find it. A function for example, it's a function next to rt_cache-add function, ip_rt_slow_route. The function is responsible for creation of a new routing cache entry. And then close to end it's page 1039 it calls rt_cache-add. Q. Okay. So it's your testimony that
2 3 4 5 6 7 8	 Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with Exhibit 2, which is the patch? A. Yes. Q. Okay. So take a look at this, and I want to ask you if this is a true and correct copy of route.c from Linux version 1.3.52. A. Yes. I think so. I cannot say it's exact. You 	2 3 4 5 6 7 8	 A. I can find it. A function for example, it's a function next to rt_cache-add function, ip_rt_slow_route. The function is responsible for creation of a new routing cache entry. And then close to end it's page 1039 it calls rt_cache-add. Q. Okay. So it's your testimony that ip_rt_slow_route calls the rt_cache-add function?
2 3 4 5 6 7 8 9	 Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with Exhibit 2, which is the patch? A. Yes. Q. Okay. So take a look at this, and I want to ask you if this is a true and correct copy of route.c from Linux version 1.3.52. A. Yes. I think so. I cannot say it's exact. You just gave me this document. 	2 3 4 5 6 7 8 9	 A. I can find it. A function for example, it's a function next to rt_cache-add function, ip_rt_slow_route. The function is responsible for creation of a new routing cache entry. And then close to end it's page 1039 it calls rt_cache-add. Q. Okay. So it's your testimony that ip_rt_slow_route calls the rt_cache-add function? A. Yes.
2 3 4 5 6 7 8 9 10	 Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with Exhibit 2, which is the patch? A. Yes. Q. Okay. So take a look at this, and I want to ask you if this is a true and correct copy of route.c from Linux version 1.3.52. A. Yes. I think so. I cannot say it's exact. You just gave me this document. Q. Okay. Mr. Kuznetsov, if you take a look at 	2 3 4 5 6 7 8 9 10	 A. I can find it. A function for example, it's a function next to rt_cache-add function, ip_rt_slow_route. The function is responsible for creation of a new routing cache entry. And then close to end it's page 1039 it calls rt_cache-add. Q. Okay. So it's your testimony that ip_rt_slow_route calls the rt_cache-add function? A. Yes. Q. And, also, could you please turn to the page
2 3 4 5 6 7 8 9 10 11	 Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with Exhibit 2, which is the patch? A. Yes. Q. Okay. So take a look at this, and I want to ask you if this is a true and correct copy of route.c from Linux version 1.3.52. A. Yes. I think so. I cannot say it's exact. You just gave me this document. Q. Okay. Mr. Kuznetsov, if you take a look at Exhibit 1, which is your declaration 	2 3 4 5 6 7 8 9 10 11	 A. I can find it. A function for example, it's a function next to rt_cache-add function, ip_rt_slow_route. The function is responsible for creation of a new routing cache entry. And then close to end it's page 1039 it calls rt_cache-add. Q. Okay. So it's your testimony that ip_rt_slow_route calls the rt_cache-add function? A. Yes. Q. And, also, could you please turn to the page ending in 1032.
2 3 4 5 6 7 8 9 10 11	 Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with Exhibit 2, which is the patch? A. Yes. Q. Okay. So take a look at this, and I want to ask you if this is a true and correct copy of route.c from Linux version 1.3.52. A. Yes. I think so. I cannot say it's exact. You just gave me this document. Q. Okay. Mr. Kuznetsov, if you take a look at Exhibit 1, which is your declaration A. Yes. 	2 3 4 5 6 7 8 9 10 11 12	 A. I can find it. A function for example, it's a function next to rt_cache-add function, ip_rt_slow_route. The function is responsible for creation of a new routing cache entry. And then close to end it's page 1039 it calls rt_cache-add. Q. Okay. So it's your testimony that ip_rt_slow_route calls the rt_cache-add function? A. Yes. Q. And, also, could you please turn to the page ending in 1032. A. Okay.
2 3 4 5 6 7 8 9 10 11 12 13	 Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with Exhibit 2, which is the patch? A. Yes. Q. Okay. So take a look at this, and I want to ask you if this is a true and correct copy of route.c from Linux version 1.3.52. A. Yes. I think so. I cannot say it's exact. You just gave me this document. Q. Okay. Mr. Kuznetsov, if you take a look at Exhibit 1, which is your declaration A. Yes. Q if you look at the second page of that, do you 	2 3 4 5 6 7 8 9 10 11 12 13	 A. I can find it. A function for example, it's a function next to rt_cache-add function, ip_rt_slow_route. The function is responsible for creation of a new routing cache entry. And then close to end it's page 1039 it calls rt_cache-add. Q. Okay. So it's your testimony that ip_rt_slow_route calls the rt_cache-add function? A. Yes. Q. And, also, could you please turn to the page ending in 1032. A. Okay. Q. Do you see a function that says rt_redirect_1 at
2 3 4 5 6 7 8 9 10 11 12 13 14	 Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with Exhibit 2, which is the patch? A. Yes. Q. Okay. So take a look at this, and I want to ask you if this is a true and correct copy of route.c from Linux version 1.3.52. A. Yes. I think so. I cannot say it's exact. You just gave me this document. Q. Okay. Mr. Kuznetsov, if you take a look at Exhibit 1, which is your declaration A. Yes. Q if you look at the second page of that, do you see paragraph 6 on the second page? 	2 3 4 5 6 7 8 9 10 11 12 13 14	 A. I can find it. A function for example, it's a function next to rt_cache-add function, ip_rt_slow_route. The function is responsible for creation of a new routing cache entry. And then close to end it's page 1039 it calls rt_cache-add. Q. Okay. So it's your testimony that ip_rt_slow_route calls the rt_cache-add function? A. Yes. Q. And, also, could you please turn to the page ending in 1032. A. Okay. Q. Do you see a function that says rt_redirect_1 at the top of the page?
2 3 4 5 6 7 8 9 10 11 12 13 14 15	 Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with Exhibit 2, which is the patch? A. Yes. Q. Okay. So take a look at this, and I want to ask you if this is a true and correct copy of route.c from Linux version 1.3.52. A. Yes. I think so. I cannot say it's exact. You just gave me this document. Q. Okay. Mr. Kuznetsov, if you take a look at Exhibit 1, which is your declaration A. Yes. Q if you look at the second page of that, do you see paragraph 6 on the second page? A. Yes. 	2 3 4 5 6 7 8 9 10 11 12 13 14 15	 A. I can find it. A function for example, it's a function next to rt_cache-add function, ip_rt_slow_route. The function is responsible for creation of a new routing cache entry. And then close to end it's page 1039 it calls rt_cache-add. Q. Okay. So it's your testimony that ip_rt_slow_route calls the rt_cache-add function? A. Yes. Q. And, also, could you please turn to the page ending in 1032. A. Okay. Q. Do you see a function that says rt_redirect_1 at the top of the page? A. Yes.
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	 Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with Exhibit 2, which is the patch? A. Yes. Q. Okay. So take a look at this, and I want to ask you if this is a true and correct copy of route.c from Linux version 1.3.52. A. Yes. I think so. I cannot say it's exact. You just gave me this document. Q. Okay. Mr. Kuznetsov, if you take a look at Exhibit 1, which is your declaration A. Yes. Q if you look at the second page of that, do you see paragraph 6 on the second page? A. Yes. Q. And it says that, Linux kernel version 1.3.52? 	2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	 A. I can find it. A function for example, it's a function next to rt_cache-add function, ip_rt_slow_route. The function is responsible for creation of a new routing cache entry. And then close to end it's page 1039 it calls rt_cache-add. Q. Okay. So it's your testimony that ip_rt_slow_route calls the rt_cache-add function? A. Yes. Q. And, also, could you please turn to the page ending in 1032. A. Okay. Q. Do you see a function that says rt_redirect_1 at the top of the page? A. Yes. Q. And do you see, around the middle of the page,
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17	 Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with Exhibit 2, which is the patch? A. Yes. Q. Okay. So take a look at this, and I want to ask you if this is a true and correct copy of route.c from Linux version 1.3.52. A. Yes. I think so. I cannot say it's exact. You just gave me this document. Q. Okay. Mr. Kuznetsov, if you take a look at Exhibit 1, which is your declaration A. Yes. Q if you look at the second page of that, do you see paragraph 6 on the second page? A. Yes. Q. And it says that, Linux kernel version 1.3.52? A. Yes. When I signed before I signed the 	2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17	 A. I can find it. A function for example, it's a function next to rt_cache-add function, ip_rt_slow_route. The function is responsible for creation of a new routing cache entry. And then close to end it's page 1039 it calls rt_cache-add. Q. Okay. So it's your testimony that ip_rt_slow_route calls the rt_cache-add function? A. Yes. Q. And, also, could you please turn to the page ending in 1032. A. Okay. Q. Do you see a function that says rt_redirect_1 at the top of the page? A. Yes. Q. And do you see, around the middle of the page, there's a statement that says rt_cache-add?
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18	 Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with Exhibit 2, which is the patch? A. Yes. Q. Okay. So take a look at this, and I want to ask you if this is a true and correct copy of route.c from Linux version 1.3.52. A. Yes. I think so. I cannot say it's exact. You just gave me this document. Q. Okay. Mr. Kuznetsov, if you take a look at Exhibit 1, which is your declaration A. Yes. Q if you look at the second page of that, do you see paragraph 6 on the second page? A. Yes. Q. And it says that, Linux kernel version 1.3.52? A. Yes. When I signed before I signed the document, I verified accurately that documents which 	2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18	 A. I can find it. A function for example, it's a function next to rt_cache-add function, ip_rt_slow_route. The function is responsible for creation of a new routing cache entry. And then close to end it's page 1039 it calls rt_cache-add. Q. Okay. So it's your testimony that ip_rt_slow_route calls the rt_cache-add function? A. Yes. Q. And, also, could you please turn to the page ending in 1032. A. Okay. Q. Do you see a function that says rt_redirect_1 at the top of the page? A. Yes. Q. And do you see, around the middle of the page, there's a statement that says rt_cache-add? A. Yes, yes, yes.
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19	 Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with Exhibit 2, which is the patch? A. Yes. Q. Okay. So take a look at this, and I want to ask you if this is a true and correct copy of route.c from Linux version 1.3.52. A. Yes. I think so. I cannot say it's exact. You just gave me this document. Q. Okay. Mr. Kuznetsov, if you take a look at Exhibit 1, which is your declaration A. Yes. Q if you look at the second page of that, do you see paragraph 6 on the second page? A. Yes. Q. And it says that, Linux kernel version 1.3.52? A. Yes. When I signed before I signed the document, I verified accurately that documents which were identified by those labels are genuine ones. 	2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19	 A. I can find it. A function for example, it's a function next to rt_cache-add function, ip_rt_slow_route. The function is responsible for creation of a new routing cache entry. And then close to end it's page 1039 it calls rt_cache-add. Q. Okay. So it's your testimony that ip_rt_slow_route calls the rt_cache-add function? A. Yes. Q. And, also, could you please turn to the page ending in 1032. A. Okay. Q. Do you see a function that says rt_redirect_1 at the top of the page? A. Yes. Q. And do you see, around the middle of the page, there's a statement that says rt_cache-add? A. Yes, yes, yes. Q. So what is happening at that line where it says
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20	 Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with Exhibit 2, which is the patch? A. Yes. Q. Okay. So take a look at this, and I want to ask you if this is a true and correct copy of route.c from Linux version 1.3.52. A. Yes. I think so. I cannot say it's exact. You just gave me this document. Q. Okay. Mr. Kuznetsov, if you take a look at Exhibit 1, which is your declaration A. Yes. Q if you look at the second page of that, do you see paragraph 6 on the second page? A. Yes. Q. And it says that, Linux kernel version 1.3.52? A. Yes. When I signed before I signed the document, I verified accurately that documents which were identified by those labels are genuine ones. Q. Okay. 	2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20	 A. I can find it. A function for example, it's a function next to rt_cache-add function, ip_rt_slow_route. The function is responsible for creation of a new routing cache entry. And then close to end it's page 1039 it calls rt_cache-add. Q. Okay. So it's your testimony that ip_rt_slow_route calls the rt_cache-add function? A. Yes. Q. And, also, could you please turn to the page ending in 1032. A. Okay. Q. Do you see a function that says rt_redirect_1 at the top of the page? A. Yes. Q. And do you see, around the middle of the page, there's a statement that says rt_cache-add? A. Yes, yes, yes. Q. So what is happening at that line where it says rt_cache_add?
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21	 Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with Exhibit 2, which is the patch? A. Yes. Q. Okay. So take a look at this, and I want to ask you if this is a true and correct copy of route.c from Linux version 1.3.52. A. Yes. I think so. I cannot say it's exact. You just gave me this document. Q. Okay. Mr. Kuznetsov, if you take a look at Exhibit 1, which is your declaration A. Yes. Q if you look at the second page of that, do you see paragraph 6 on the second page? A. Yes. Q. And it says that, Linux kernel version 1.3.52? A. Yes. When I signed before I signed the document, I verified accurately that documents which were identified by those labels are genuine ones. Q. Okay. A. But I just you didn't tell me that it is that 	2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21	 A. I can find it. A function for example, it's a function next to rt_cache-add function, ip_rt_slow_route. The function is responsible for creation of a new routing cache entry. And then close to end it's page 1039 it calls rt_cache-add. Q. Okay. So it's your testimony that ip_rt_slow_route calls the rt_cache-add function? A. Yes. Q. And, also, could you please turn to the page ending in 1032. A. Okay. Q. Do you see a function that says rt_redirect_1 at the top of the page? A. Yes. Q. And do you see, around the middle of the page, there's a statement that says rt_cache-add? A. Yes, yes, yes. Q. So what is happening at that line where it says rt_cache_add? A. Rt_cache-add adds new cache entries, so it's
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22	 Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with Exhibit 2, which is the patch? A. Yes. Q. Okay. So take a look at this, and I want to ask you if this is a true and correct copy of route.c from Linux version 1.3.52. A. Yes. I think so. I cannot say it's exact. You just gave me this document. Q. Okay. Mr. Kuznetsov, if you take a look at Exhibit 1, which is your declaration A. Yes. Q if you look at the second page of that, do you see paragraph 6 on the second page? A. Yes. Q. And it says that, Linux kernel version 1.3.52? A. Yes. When I signed before I signed the document, I verified accurately that documents which were identified by those labels are genuine ones. Q. Okay. A. But I just you didn't tell me that it is that document. 	2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22	 A. I can find it. A function for example, it's a function next to rt_cache-add function, ip_rt_slow_route. The function is responsible for creation of a new routing cache entry. And then close to end it's page 1039 it calls rt_cache-add. Q. Okay. So it's your testimony that ip_rt_slow_route calls the rt_cache-add function? A. Yes. Q. And, also, could you please turn to the page ending in 1032. A. Okay. Q. Do you see a function that says rt_redirect_1 at the top of the page? A. Yes. Q. And do you see, around the middle of the page, there's a statement that says rt_cache-add? A. Yes, yes, yes. Q. So what is happening at that line where it says rt_cache_add? A. Rt_cache-add adds new cache entries, so it's called when we have to add new cache entry. Function
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23	 Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with Exhibit 2, which is the patch? A. Yes. Q. Okay. So take a look at this, and I want to ask you if this is a true and correct copy of route.c from Linux version 1.3.52. A. Yes. I think so. I cannot say it's exact. You just gave me this document. Q. Okay. Mr. Kuznetsov, if you take a look at Exhibit 1, which is your declaration A. Yes. Q if you look at the second page of that, do you see paragraph 6 on the second page? A. Yes. Q. And it says that, Linux kernel version 1.3.52? A. Yes. When I signed before I signed the document, I verified accurately that documents which were identified by those labels are genuine ones. Q. Okay. A. But I just you didn't tell me that it is that document. 	2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23	 A. I can find it. A function for example, it's a function next to rt_cache-add function, ip_rt_slow_route. The function is responsible for creation of a new routing cache entry. And then close to end it's page 1039 it calls rt_cache-add. Q. Okay. So it's your testimony that ip_rt_slow_route calls the rt_cache-add function? A. Yes. Q. And, also, could you please turn to the page ending in 1032. A. Okay. Q. Do you see a function that says rt_redirect_1 at the top of the page? A. Yes. Q. And do you see, around the middle of the page, there's a statement that says rt_cache-add? A. Yes, yes, yes. Q. So what is happening at that line where it says rt_cache_add? A. Rt_cache-add adds new cache entries, so it's called when we have to add new cache entry. Function rt_redirect_1 creates stockhold redirected entry.
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22	 Q. Okay. And is the portion that you wrote the portion that we discussed earlier in connection with Exhibit 2, which is the patch? A. Yes. Q. Okay. So take a look at this, and I want to ask you if this is a true and correct copy of route.c from Linux version 1.3.52. A. Yes. I think so. I cannot say it's exact. You just gave me this document. Q. Okay. Mr. Kuznetsov, if you take a look at Exhibit 1, which is your declaration A. Yes. Q if you look at the second page of that, do you see paragraph 6 on the second page? A. Yes. Q. And it says that, Linux kernel version 1.3.52? A. Yes. When I signed before I signed the document, I verified accurately that documents which were identified by those labels are genuine ones. Q. Okay. A. But I just you didn't tell me that it is that document. 	2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22	 A. I can find it. A function for example, it's a function next to rt_cache-add function, ip_rt_slow_route. The function is responsible for creation of a new routing cache entry. And then close to end it's page 1039 it calls rt_cache-add. Q. Okay. So it's your testimony that ip_rt_slow_route calls the rt_cache-add function? A. Yes. Q. And, also, could you please turn to the page ending in 1032. A. Okay. Q. Do you see a function that says rt_redirect_1 at the top of the page? A. Yes. Q. And do you see, around the middle of the page, there's a statement that says rt_cache-add? A. Yes, yes, yes. Q. So what is happening at that line where it says rt_cache_add? A. Rt_cache-add adds new cache entries, so it's called when we have to add new cache entry. Function

	Page 62		Page 64
1	Is that the same while statement that we	1	(Short recess taken.)
2	discussed earlier in connection with your patch?	2	THE VIDEOGRAPHER: On the record. This
3	A. Yes.	3	is cassette No. 3, and the time is 4:35.
4	Q. Okay. And so is this a while loop?	4	Sorry. 3:35.
5	A. Yes.	5	Q. Okay. Mr. Kuznetsov, thank you. We are back on
6	Q. And can you identify the beginning and ending of	6	the record.
7	that while loop?	7	And shortly before we took our break, you we
8	A. Yes.	8	were discussing the while loop contained on page 1037,
9	Q. Okay. Please do.	9	in rt cache-add.
10	A. Beginning is a line which starts from the word	10	If you look at the line below the cli in the
11	While. It's a line which starts from word While. It	11	while loop, the if statement, could you what does
12	immediately follows comment which we discussed which	12	that if statement do?
13	starts cleanup duplicated and aged off entries.	13	A. This statement checks the current routing cache
14	And this loop ends right before line containing	14	entry can be deleted. It's either expired or duplicate.
15	restore flags.	15	Q. Okay. And if the record is expired, what happens
16	Q. Just so the record is clear, the entire while	16	next?
17	loop is contained on page 1037, correct?	17	A. If the record is expired, it is deleted from hash
18	A. Yes.	18	chain. Routing cache size is decreased. Then it's
19	Q. And so does this while loop continue walking	19	destroyed, and we proceed with current table of the hash
20	through the records in the linked list that was accessed	20	chain.
21	earlier?	21	Q. Very good. And does any of the lines of code in
22	A. Yes.	22	the while loop go back to the head of the linked list?
23	Q. And so does the while loop start over at the	23	A. No.
24	beginning of the linked list?	24	(Exhibit No. 4 marked for
25	A. Not it continues scan the linked list.	25	identification.)
	Page 63		Page 65
1	-	1	-
1	Current head of linked list is already new. So we	1	Q. I would like to introduce another exhibit. This
2	have this While continues scan through linked list,	2	is Exhibit 4.
3	but new entries.	3	Mr. Kuznetsov, I'm handing you what's been marked
4	We don't want to delete these entries, so we	4	as Exhibit 4 (indicating). It is if you go by the
5	continue the scan of table of linked list.	5	number at the bottom of the page, it starts with DEF.
6	Q. Okay. Can you describe what happens within this	6	It starts with DEF00009321
7 8	while loop?	7	A. Yes.
	A. Yes. We check we scan through all the chain,	8	Q and continues through DEF00009349.
9 10	and analyze each routing cache entry which we encounter for the full condition.	9 10	A. Yes.
11	First, we check that this entry is too old and	1	Q. Do you recognize this document?A. Yes. It is the same file. Probably it is of
12	not a reference right now. In this case, we can delete	11 12	according to the number before
13		13	(Witness peruses documents.)
14	this entry. This entry is expired. And also we check that this entry points to the same destination which	14	A. I don't see reference to this number.
14	to the same destination as entry which we have just	15	Q. Let's also use let's use the bottom number
16	added. So we must delete this entry.	16	then that says KS-DEF. And it starts with
17	So if one of these conditions is satisfied, we	17	A. Page 6 yes I don't see it. Oh, yes. I
18	unlink current entry from hash chain, decrease size of	18	found it. It's statement No. 3 in my declaration. Yes,
19	cache size to follow the fact that we deleted some	19	it is the same file.
20	entry, and destroy this routing entry.	20	Q. And what is this document?
21	The function rtfree destroys current entry, and	21	A. It is the same file.
22	after this we continue the scanning table for link.	22	Q. So is it route.c?
23	Q. Okay.	23	A. Yes.
	· ·	1	
24	THE VIDEOGRAPHER: This is the end of	124	O. And which version of Linux?
24 25	THE VIDEOGRAPHER: This is the end of tape No. 2. Off the record. Time is 3:22.	24 25	Q. And which version of Linux?A. It's from kernel 1.3.42.

1 Q. Okay. And is this a true and correct copy of the 1 Q. But 2 A. Yes. Identified this. Q. Okay. And does this does Exhibit 4, this Q. Okay. And you downloaded that file in 1955? 4 Q. Okay. And does this does Exhibit 4, this Q. Okay. And you downloaded that file in 1955? 4 A. Yes. A. Yes. A. Yes. 8 Q. And iryou turn to the page ending in 708 and ends with 708, do you see that page? O. Okay. Very good. Let's introduce another 1 A. Yes. T. Cache-add? Imathing you what has been marked as 1 A. Yes. T. Cache-add? Imathing you what has been marked as 1 A. Yes. T. Cache-add? Imathing you what has been marked as 1 A. Yes. T. Cache-add? T. Mado you recognize this document? 1 A. Yes. T. Yes. A. I's almost the same of cache-add function that 1 A. Yes. T. Yes. A. I's almost the same of cache-add function a little 2 Q. Okay. Do you know when this version of route. T. Yes. A. I's almost the same docorrect cory? 2 Q. Okay. Do you know when this version of route. T. Yes. A. Yes. It was publicly available as Linux version <th></th> <th>Page 66</th> <th></th> <th>Page 68</th>		Page 66		Page 68
2 route from abov? 2 A. Theow it was thousands of people. 3 A. Yes. I identified this. 2 Q. Okay. And you downloaded that file in 1995? 4 Q. Okay. And you downloaded that file in the path that was Exhibit 2 today? 4 A. Yes. 7 A. Yes. Particularly 1 downloaded exactly at the sing the bottom set of numbers, it starts with KS-DEF 9 4 10 and ends with 708, do you see that page? 10 0. I am handing you what has been marked as 11 A. Yes. 11 2 0. And do you see, near the top of that page. 12 Q. And is this the samer t_ eache-add function that 14 A. Yes. 17 0. I am handing you what has been marked as 13 A. Yes. 16 A. Yes. 17 0. I am handing you what has been marked as 14 A. Yes. 16 A. Yes. 17 0. I am handing you what has been marked as 14 A. Yes. 17 0. And is this the same t_ eache-add function that 16 A. Yes. 13 A. Yes. 13.42 Q. Okay. Do you know when this version of route.cwas 13 14 14 12 13 Q. So yins tom make the	1	Q. Okay. And is this a true and correct copy of the	1	Q. But
3 A. Yes. Lidentified this. 3 Q. Okay. And you downloaded that filt in 1995? 4 Q. Okay. And you downloaded that filt in 1995? A. Yes. Particularly I downloaded exactly at the 5 day when it was published. 6 included in the patch that was Exhibit 2 today? A. Yes. 8 Q. And if you turn to the page ending in 708 using the bottom set of numbers, it starts with KS-DEF 9 10 and ends with 708, do you see, near the top of that page? 10 Q. Iam handing you what has been marked as 11 A. Yes. 11 Exhibit 5. It is - left see. 12 12 Q. And is do you see, near the top of that page? 13 13 A. Yes. 13 T. eache-add? 13 14 A. Yes. 14 A. Yes. 13 14 A. Yes. 15 Q. And is this the same rt_eache-add function that 15 And do you recognize this document? 16 we saw in your patch that was Exhibit 2 in today? 16 A. Yes. 17 17 Q. Okay. Do you know when this version of route. was 16 A. Yes. 17 13 a. Yes. It was publicly available as Linux version 13.51. 13.52 2 Q. S	2		2	
4 Q. Okay. And does this - does Exhibit 4, this 4 A. Yes. Particularly I downloaded exactly at the 5 version of route, included in the patch that was Exhibit 2 today? 6 Q. Okay. Very good. Let's introduce another 7 A. Yes. Q. And if you turn to the page ending in 708 9 9 9 0 And if you turn to the page ending in 708 9 0 Q. Ana it was published. 6 Q. Okay. Very good. Let's introduce another 9 and ends with 708, do you see that page, 11 1 1 0 1 1 0 Q. Ana it was published. 1 0 0 0 Q. Iam handing you what has been marked as 11 A. Yes. 1 3 0 And was this beamert_cache-add function that 1 1 DEF00007899. 1 1 3 1 4 A Yes. 12 Q. And is this the same rt_cache-add function that 1 1 1 1 4 4 4 4 4 4 4 4 4 4 4 4 5 5 6 6 0 1 4 4 4 4	3	A. Yes. I identified this.		
5 version of route, c, include the code you wrote that was included in the patch that was Exhibit 2 today? A. Yes. Q. And if you turn to the page ending in 708 is day when it was published. Q. Okay. Very good. Let's introduce another 8 Q. And if you turn to the page ending in 708 is distribut. C. Okay. Very good. Let's introduce another 9 and ends with 708, do you see that page? 0. I am handing you what has been marked as 11 A. Yes. 11 Exhibit 5. It is let's see. 12 Q. And do you see, near the top of that page, 12 If we use the numbers at the bottom right corner 13 rt cache-add? 13 DF00007895 and do you recognize this document? 14 A. Yes. 6 A. Yes. 17 Q. And what is it? 15 Q. And, by our ache with this version of route. was 17 Q. And was this decument. It's different 15 P. Okay. Do you know when this version of route. 18 A. Yes. it was publicly available as Linux version 16 A. Yes. 1. Stall 20 So is this a true and correct copy? 17 Q. Okay. Do you know when this version of route. 21 version 1.3.51. 20 Okay. Do you kn	4	Q. Okay. And does this does Exhibit 4, this	4	
6 included in the patch that was Exhibit 2 today? 6 Q. Okay. Very good. Let's introduce another exhibit. 7 A. Yes. (Exhibit No. 5 marked for identification.) (Exhibit No. 5 marked for identification.) 9 using the bottom set of numbers, it starts with KS-DEF 9 (Exhibit S. It is - let's see. 12 Q. And do you see, near the top of that page, 1 1 (Impace, it starts with DEFO007865 and ends at 14 14 A. Yes. 1 Mesaw in your patch that was Exhibit 2 in today's 1 1 15 Q. And is this the same rt. cache-add function that 1 1 1 A. Yes. 16 A. Yes. 1 DEF00007899. 1 1 16 A. Yes. 1 A. Yes. 1 1 17 deposition? 1 1 A. Mes out now when this version of route. caws 1 1 1 20 News. braublichy available as Linux version 1.3.42. 2 Q. So is this a true and correct copy? 2 3 A. Yes. it is ruce accurate copy? 3 A. Yes. 1 2 2 Q. So is this a true and correct copy? 3 A. Yes. it is annost the same rt_ca	5		5	
7 A. Yes. 7 exhibit. 9 using the bottom set of numbers, it starts with KS-DEF 9 identification.) 10 and ends with 708, do you see that page? 10 Q. I am handing you what has been marked as 12 Q. And do you see, near the top of that page. 11 Exhibit 5. It is - 1ef's see. 12 Q. And do you see, near the top of that page. 12 If we use the numbers at the bottom right corner 14 A. Yes. 12 Fwe use the numbers at the bottom right corner 14 A. Yes. 12 If we use the numbers at the bottom right corner 15 Q. And is this the same rt_cache-add function that 15 A. Yes. 16 A. Yes. 12 A. Yes. 13 17 Q. Okay. Do you know when this version of route.c 14 A. Yes. 14 14 20 made publicly available as Linux version 12 version 1.3.51. 20 0. So is this a true and correct copy? 21 versified before that it was issued on November 17 of 19 you see rt_cache-add at the top? 2 21 versified before that it was issued on November 16, 1992. 2 0. Okay. And t	6	-		
9 using the bottom set of numbers, it starts with KS-DEF 9 identification.) 10 and ends with 708, do you see that page? 10 Q. I am handing you what has been marked as 11 A. Yes. 11 Exhibit 5. It is - let's see. 12 Q. And do you see, near the top of that page, 12 Exhibit 5. It is - let's see. 13 rt cache-add? 13 of the page, it starts with DEF00007865 and ends at 14 DEF00007890 16 A. Yes. 15 Q. And is this the same rt_cache-add function that 16 A. Yes. 16 A. Yes. 16 A. Yes. 17 Q. And what is it? 16 A. Yes. 18 A. Yes. It was publicly available as any time? 16 A. Yes. 12 Page 67 Verified before that it was issued on November 17 of 21 22 Q. Okay. Do you know when this version of route. 22 Q. Okay. And turning to the page ending in 7892, 14 verified before that it was issued on November 17 of 19 25 Q. Okay. And was this document ever publicly available? 14 available on? 1994. 1944. A. Yes. 30 Q	7		7	
10 and ends with 708, do you see that page? 10 Q. I am handing you what has been marked as 11 A. Yes. 11 2 Q. And do you see, near the top of that page, 12 13 rt_ cache-add? 13 14 A. Yes. 13 15 Q. And is this the same rt_cache-add function that 15 16 we saw in your patch that was Exhibit 2 in today's 16 16 we saw in your patch that was Exhibit 2 in today's 16 17 Q. And what is it? 18 18 A. Yes. 17 Q. And what is it? 18 A. Yes. 18 A. It's almost the same document. It's different 19 Q. Okay. Do you know when this version of route. was 19 previous one in few lines. It's document form a little 11 yersion 1.3.51. 22 Q. So is this a true and correct copy? 23 24 was made publicly available? 24 route. c from Linux version 1.3.51. 25 25 A. Actually. I cannot tell this exactly. But I 25 Q. Okay. And turning to the page ending in 7892, 24 was made publicly available? 1 yous	8	Q. And if you turn to the page ending in 708	8	(Exhibit No. 5 marked for
11A. Yes.11Exhibit 5. It is left see.12Q. And do you see, near the top of that page, rt cache-add?12If we use the numbers at the bottom right corner14A. Yes.13of the page, it starts with DEF00007859.14A. Yes.14DEF00007899.15Q. And is this the same rt cache-add function that15And do you recognize this document?16we saw in your patch that was Exhibit 2 in today's16A. Yes.17deposition?17Q. And what is it?18A. Yes. Exactly the same.19previous one in few lines. It's different19Q. Okay. Do you know when this version of route.c was19previous one in few lines. It's different21A. Yes. It was publicly available as Linux version212122Q. So is this a true and correct copy?23Q. Okay. Do you know when this version of orute.c2423Q. Okay. Do you know when this version of route.c24route. from Linux version 1.3.51.24verified before that it was issued on November 17 of2A. Yes.25A. Actually, I cannot tell this exactly. But I25A. Yes.3Q. So yust to make the record clear, this version of1you see rt_cache-add at the top?3Q. So just to make the record clear, this version of2A. Yes.4A. So, just to make the record clear, this version of3Q. Okay. And is this the same rt_cache-add function4this document in 1995?4A. Yes.1	9	using the bottom set of numbers, it starts with KS-DEF	9	identification.)
12 Q. And do you see, near the top of that page, it cache-add? 12 If we use the numbers at the bottom right comer of the page, it starts with DEP00007865 and ends at DEP00007869. 13 A. Yes. 14 DEP00007899. 15 Q. And is this the same rt_cache-add function that 15 And do you recognize this document? 16 we saw in your patch that was Exhibit 2 in today's 16 A. Yes. 16 we saw in your patch that was Exhibit 2 in today's 17 Q. And what is it? 18 A. Yes. Standly the same. 19 Q. Okay. Do you know if this version of route. cwas 19 21 1.3.42. 20 Iater kernel version, as I testified before, kernel 20 22 1.3.42. 22 Q. So is this a true and correct copy? 2. 23 Q. Okay. Do you know when this version of route. 20 Nersi is true, true accurate copy of file 20 24 was made publicly available? 24 You see rt_cache-add at the top? 2 2 Q. Okay. And what year - excuse me. What date was it 3 Q. Okay. And was this document ever publicly 3 Q. So just to make the record clear, this version of route. 3 Q. Okay. And was this document ever pu	10	and ends with 708, do you see that page?	10	Q. I am handing you what has been marked as
13 rt. cache-add? 13 of the page, it starts with DEF00007865 and ends at 14 A. Yes. 14 DEF00007899. 16 we saw in your patch that was Exhibit 2 in today's 16 And do you recognize this document? 17 deposition? 17 Q. And what is it? 18 A. Yes. Exactly the same. 18 A. It's almost the same document. It's different 19 Q. Okay. Do you know if this version of route.c was 19 previous one in few lines. It's document from a little 21 A. Yes. It was publicly available as Linux version 21 version 1.3.51. 22 Q. Okay. Do you know when this version of route.c 22 Q. So is this a true and correct copy? 23 Q. Okay. Do you know when this version of route.c 24 version 1.3.51. 25 24 was made publicly available? 24 Q. Okay. And utruing to the page ending in 7892, 24 25 A. Actually, I cannot tell this exactly. But I 25 Q. Okay. And is this the same rt_cache-add function that is in Exhibit 2? 2 A. Yes. 1 verified before that it was issued on November 17 of 2 A. Yes. 9 Q. Okay. And is this the same rt_cache-add function t	11	A. Yes.	11	Exhibit 5. It is let's see.
14 Ā. Yes. 14 DEF00007899. 15 Q. And is this the same rt_cache-add function that 15 And do you recognize this document? 15 we saw in your patch that was Exhibit 2 in today's 16 A. Yes. 17 deposition? 17 Q. And what is it? 18 A. Yes. Exactly the same. 17 Q. And what is it? 19 O. Okay. Do you know if this version of route.c was 18 A. It's almost the same document. It's different 21 A. Yes. It was publicly available at any time? 20 So is this a true and correct copy? 23 Q. Okay. Do you know when this version of route.c 23 Q. So is this a true accurate copy of file 24 was made publicly available? 21 22 Q. Okay. And turning to the page ending in 7892, 24 was made publicly available? 24 25 Q. Okay. And turning to the page ending in 7892, 25 A. Actually, I cannot tell this exactly. But I 25 So what year - excuse me. What date was it 3 4 available on? 2 4. Yes. 9 O. Okay. And is this the same rt_cache-add function 4 available on? 2 A. Yes. </td <td>12</td> <td>Q. And do you see, near the top of that page,</td> <td>12</td> <td>If we use the numbers at the bottom right corner</td>	12	Q. And do you see, near the top of that page,	12	If we use the numbers at the bottom right corner
15 Q. And is this the same rt_cache-add function that 15 And do you recognize this document? 16 we saw in your patch that was Exhibit 2 in today's 16 A. Yes. 18 A. Yes. Exactly the same. 16 A. Yes. 19 Q. Okay. Do you know if this version of route. was 19 previous one in few lines. It's document form a little 19 a. Yes. It was publicly available at any time? 10 later kernel version, as I testified before, kernel 21 A. Yes. It was publicly available as Linux version 20 So is this a true and correct copy? 23 Q. Okay. Do you know when this version of route. 23 A. Yes, it is true, true accurate copy of file 24 was made publicly available? 24 Yes. Page 67 24 was inside on November 17 of 2 Q. Okay. And turning to the page ending in 7892, 25 A. Actually, I cannot tell this exactly. But I 25 Q. Okay. And si this the same rt_cache-add function 4 available on? 4 you sec rt_cache-add at the top? 2 1994. November 16. 29 A. Yes. 3 Q. Okay. And was this document ever publicly 7 A. So: I'm snory. 1995.	13	rt_cache-add?	13	of the page, it starts with DEF00007865 and ends at
16 w. saw in your patch that was Exhibit 2 in today's 16 A. Yes. 17 deposition? 17 Q. And what is it? 18 A. Yes. Exactly the same. 17 Q. And what is it? 20 made publicly available at any time? 18 A. It's almost the same document. It's different previous one in few lines. It's document from a little 21 A. Yes. It was publicly available as Linux version 21 version 1.3.51. 22 Q. Okay. Do you know when this version of route.c 23 A. Yes, it is true accurate copy of file 23 Q. Okay. Do you know when this version of route.c 24 was made publicly available? 25 24 was made publicly available? 25 A. Actually, I cannot tell this exactly. But I 25 Q. Okay. And turning to the page coding in 7892, 2 1 verified before that it was issued on November I? of 1 you see rt_cache-add at the top? 2 A. Yes. 3 Q. So what year excuse me. What date was it 3 Q. Okay. And is this the same rt_cache-add function 4 available on? 4 that is in Exhibit 2? 5 A. Yes. 4 Q. So just to make the record clear, this version of 9	14	A. Yes.	14	DEF00007899.
17 deposition? 17 Q. And what is it? 18 A. Yes. Exactly the same. 18 A. It's almost the same document. It's different 19 Q. Okay. Do you know if this version of route. was 18 A. It's almost the same document from a little 20 made publicly available at any time? 20 I.3.42. 21 21 I.3.42. 23 Q. Okay. Do you know when this version of route. 23 A. Yes, it is true, true accurate copy of file 24 was made publicly available? 23 A. Actually, I cannot tell this exactly. But I 25 Q. Okay. And turning to the page ending in 7892, 25 A. Actually, I cannot tell this exactly. But I 25 Q. Okay. And turning to the page ending in 7892, 2 you see rt_cache-add at the top? 2 A. Yes. 2 1 you see rt_cache-add at the top? 2 A. Yes. 3 Q. So what year - excuse me. What date was it 3 Q. Okay. And is this the same rt_cache-add function 4 available on? A. Yes. 4 Yes. 4 7 A. So just to make the record clear, this version of 9 0. Okay. And is this the same at_cache-add function 4	15		15	And do you recognize this document?
18 Å. Yes. Exactly the same. 18 Å. It's almost the same document. It's different previous one in few lines. It's document from a little later kernel version, as I testified before, kernel 19 Q. Okay. Do you know if this version of route.c 19 21 A. Yes. It was publicly available as Linux version 20 22 Q. Okay. Do you know when this version of route.c 21 23 Q. Okay. Do you know when this version of route.c 22 24 was made publicly available? 23 25 A. Actually, I cannot tell this exactly. But I 25 26 Page 67 Page 67 29 Q. So what year - excuse me. What date was it 3 3 Q. So what year - excuse me. What date was it 3 4 available on? 2 A. Yes. 7 A. 5. I'm sorry. 1995. 8 A. Yes. 8 A. Se. 1 was publicly available on November 16, 1995? 9 9 route.c was publicly available on November 16, 1995? 9 1.3.51. 10 Q. And how and how could someone have accessed 1 A. Ore around December December 27, 1995. 11 Q. And how and how could someone have acc			16	
19 Q. Okay. Do you know if this version of route.c was made publicly available at any time? 19 previous one in few lines. It's document from a little 20 made publicly available at any time? 20 later Kernel version, as I testified before, kernel 21 A. Yes. It was publicly available as Linux version 21 Version 1.3.51. 21 1.3.42. Q. Okay. Do you know when this version of route.c Q. So is this a true and correct copy? 23 Q. Okay. Do you know when this exactly. But I 25 A. Actually, I cannot tell this exactly. But I 25 25 A. Actually, I cannot tell this exactly. But I 25 Q. Okay. And turning to the page ending in 7892, 26 Page 67 Page 67 Page 69 27 verified before that it was issued on November 17 of available on? 1 you see rt_cache-add at the top? 2 A. Yes. 3 Q. So what year - excuse me. What date was it available on? 2 A. Yes. 3 Q. Okay. And is this the same rt_cache-add function 4 that is in Exhibit 2? 4 available on? 4. Yes. 4. Yes. 4. Yes. 4. Yes. 1 9 1.3.51. 5 A. November 16. 99.5? 1. A. On or around December - De		*	17	
20 made publicly available at any time? 20 later kernel version, as I testified before, kernel 21 A. Yes. It was publicly available as Linux version 21 version 1.3.51. 22 Q. Okay. Do you know when this version of route.c. 23 A. Yes, it is true, true accurate copy of file 24 was made publicly available? 24 7. Yes, it is true, true accurate copy of file 25 A. Actually, I cannot tell this exactly. But I 25 Q. Okay. And turning to the page ending in 7892, 26 Page 67 Page 69 27 Yerified before that it was issued on November 17 of 1 you see rt_cache-add at the top? 28 Q. So what year excuse me. What date was it 3 Q. Okay. And is this the same rt_cache-add function 4 available on? 5 A. Yes. 3 Q. Okay. And was this document ever publicly 7 A. Se. 9 O. Okay. And was this document ever publicly 3 A. Yes. 8 Q. So yust to make the record clear, this version of 9 A. Yes. 10 Q. And was this document ever publicly 9 O. And how and how could someone have accessed 11 A. On or arounol December - December 27, 1995.				
21 A. Yes. It was publicly available as Linux version 21 version 1.3.51. 22 Q. Okay. Do you know when this version of route.c 23 A. Yes, it is true, true accurate copy? 23 Q. Okay. Do you know when this version of route.c 23 A. Yes, it is true, true accurate copy? 24 was made publicly available? 24 Yes, it is true, true accurate copy? 25 A. Actually, I cannot tell this exactly. But I 25 Q. Okay. And turning to the page ending in 7892, 7 Page 67 Page 67 Page 69 1 you see rt_cache-add at the top? A. Yes. 3 Q. So what year excuse me. What date was it 3 Q. Okay. And is this the same rt_cache-add function 4 that is in Exhibit 2? A. Yes. 4 Yes. 6 Q. 1994? 5 A. Yes. 4 7 A. So just to make the record clear, this version of 9 13.51. 2 9 route.c was publicly available on November 16, 1995? 7 available? 3 A. On or around DecemberDecember 27, 1995. 10 Q. And how and how could someone have accessed 1 A. On or around DecemberDecember 27, 1995.				-
22 1.3.42. Q. Okay. Do you know when this version of route.c 22 Q. So is this a true and correct copy? 23 Q. Okay. Do you know when this version of route.c 23 A. Yes, it is true, true accurate copy of file 24 was made publicly available? 24 A. Yes, it is true, true accurate copy? 25 A. Actually, I cannot tell this exactly. But 1 25 Q. Okay. And turning to the page ending in 7892, 26 Page 67 Page 67 Q. Okay. And turning to the page ending in 7892, 21 1994. November 16. 2 A. Yes. 3 Q. So what year excuse me. What date was it 3 Q. Okay. And is this the same rt_cache-add function 4 available on? 4 4. Yes. 3 Q. Okay. And was this document ever publicly 7 A. 5. I'm sorry. 1995. 6 Q. Okay. And was this document ever publicly 3 8 Q. So just to make the record clear, this version of 9 1.3.51. Q. And when was it first publicly available? 1 Q. And how and how could someone have accessed 1 A. On or around December December 27, 1995. 1 Q. And how and how could someone have accessed 1 A. Could unload this documen				
23 Q. Okay. Do you know when this version of route. 23 A. Yes, it is true, true accurate copy of file 24 was made publicly available? 24 25 A. Actually, I cannot tell this exactly. But I 25 Page 67 Page 67 1 verified before that it was issued on November 17 of 1 2 1.994. November 16. 2 3 Q. So what year excuse me. What date was it 3 4 available on? 3 5 A. November 16, 1994. 5 6 Q. 1994? 7 7 A. S. I'm sorry. 1995. 8 8 Q. So just to make the record clear, this version of 9 9 route. c was publicly available on November 16, 1995? 1 10 A. Yes. 10 11 Q. And how - and how could someone have accessed 10 12 this document in 1995? 11 13 A. He could go to FTP site where Linux kernels are 13 14 stored to download huge file which contains all the 14 15 files related to Linux kernels. 15 16		- · ·		
24 was made publicly available? 24 route.c from Linux version 1.3.51. 25 A. Actually, I cannot tell this exactly. But I 25 Q. Okay. And turning to the page ending in 7892, Page 67 Page 67 Page 69 1 verified before that it was issued on November 17 of 1 you see rt_cache-add at the top? 2 A. November 16. 2 A. Yes. 3 Q. So what year excuse me. What date was it 3 Q. Okay. And is this the same rt_cache-add function 4 available on? 5 A. Yes. Q. Okay. And was this document ever publicly 7 A. S. I'm sorry. 1995. 6 Q. Okay. And was this document ever publicly 8 Q. So just to make the record clear, this version of 9 13.51. 9 route.c from 10 Q. And was this document ever publicly available? 11 Q. And how and how could someone have accessed 11 A. Or or around December December 27, 1995. 12 this document in 195? 12 Q. And how could someone have accessed this document 13 A. He could go to FTP site where Linux kernels are 13 in 1995? 14 stored to download huge file				
25 A. Actually, I cannot tell this exactly. But I 25 Q. Okay. And turning to the page ending in 7892, Page 67 Page 69 1 verified before that it was issued on November 17 of 1 you see rt_cache-add at the top? 2 1994. November 16. 2 A. Yes. 3 Q. So what year excuse me. What date was it 3 Q. Okay. And is this the same rt_cache-add function 4 available on? 5 A. Yes. 3 5 A. November 16, 1994. 5 A. Yes. 6 Q. 1994? 6 Q. Okay. And was this document ever publicly 7 A. 5. I'm sorry. 1995. 8 A. Yes. 8 Q. So just to make the record clear, this version of 9 13.51. 9 route.c was publicly available on November 16, 1995? 10 Q. And when was it first publicly available? 11 Q. And how and how could someone have accessed 11 A. On or around December December 27, 1995. 12 this document in 1995? 12 Q. And who wanyone who accessed this document 13 A. He could go to FTP site where Linux kernels are 13 in 1995? 14 <t< td=""><td></td><td></td><td></td><td></td></t<>				
Page 67 Page 67 1 verified before that it was issued on November 17 of 1 you see rt_cache-add at the top? 2 1994. November 16. 2 A. Yes. 3 Q. So what year excuse me. What date was it 3 Q. Okay. And is this the same rt_cache-add function 4 available on? 4 that is in Exhibit 2? 5 5 A. November 16, 1994. 6 Q. Okay. And was this document ever publicly 7 A. 5. I'm sorry. 1995. 7 available on November 16, 1995? 8 Q. So just to make the record clear, this version of 8 A. Yes. 9 route.c was publicly available on November 16, 1995? 9 1.3.51. 10 A. Yes. 10 Q. And when was it first publicly available? 11 Q. And how and how could someone have accessed 11 A. Or or around December December 27, 1995. 12 this document in 1995? 12 Q. And whow was it first publicly available? 13 A. He could go to FTP site where Linux kernels are 13 in 1995? 14 stored to download huge file which contains all the 14 A. Could unload this document as part of Linux				
1verified before that it was issued on November 17 of1you see rt_cache-add at the top?21994. November 16.2A. Yes.3Q. So what year excuse me. What date was it3Q. Okay. And is this the same rt_cache-add function4available on?4that is in Exhibit 2?5A. November 16, 1994.5A. Yes.6Q. 1994?6Q. Okay. And was this document ever publicly7A. 5. I'm sorty. 1995.6Q. Okay. And was this document ever publicly8Q. So just to make the record clear, this version of8A. Yes. It was publicly available as part of kernel9route.c was publicly available on November 16, 1995?91.3.51.10A. Yes.10Q. And when was it first publicly available?11Q. And how and how could someone have accessed11A. Ora ora unad December December 27, 1995.12this document in 1995?12Q. And how could someone have accessed this document13A. He could go to FTP site where Linux kernels are13in 1995?14stored to download huge file which contains all the14A. Could unload this document as part of Linux15files related to Linux kernels.15source archive version by No. 1.3.52, after unloading16This file was numbered 1.3.42, and you could16the FTP site.17Q. Okay. Do you know anyone who accessed this file19A. Yes. It was me; Linus Torvalds, who created the19Q. Okay. Do you know anyone who accessed	25	A. Actually, I cannot tell this exactly. But I	25	Q. Okay. And turning to the page ending in 7892,
21994. November 16.2A. Yes.3Q. So what year excuse me. What date was it3Q. Okay. And is this the same rt_cache-add function4available on?4that is in Exhibit 2?5A. November 16, 1994.5A. Yes.6Q. 1994?5A. Yes.7A. 5. I'm sorry. 1995.6Q. Okay. And was this document ever publicly7A. 5. I'm sorry. 1995.78Q. So just to make the record clear, this version of8A. Yes.9route.c was publicly available on November 16, 1995?91.3.51.10A. Yes.10Q. And when was it first publicly available?11Q. And how and how could someone have accessed11A. On or around December December 27, 1995.12this document in 1995?12Q. And how could someone have accessed13A. He could go to FTP site where Linux kernels are13in 1995?14stored to download huge file which contains all the14A. Could unload this document as part of Linux15files related to Linux kernels.15source archive version by No. 1.3.52, after unloading16This file was numbered 1.3.42, and you could16the FTP site.17Q. Okay. Do you know anyone who accessed this file19A. Yes. It was me; Linus Torvalds, who created the20in 1995?20document, at least Alan Cox, to whom this patch was21A. The question is more complicated. Yes, of20Okay.22		Page 67		Page 69
3Q. So what year excuse me. What date was it3Q. Okay. And is this the same rt_cache-add function4available on?4that is in Exhibit 2?5A. November 16, 1994.6Q. Okay. And was this document ever publicly7A. 5. I'm sorry. 1995.6Q. Okay. And was this document ever publicly8Q. So just to make the record clear, this version of8A. Yes. It was publicly available as part of kernel9route.c was publicly available on November 16, 1995?91.3.51.10A. Yes.10Q. And when was it first publicly available?11Q. And how and how could someone have accessed11A. On or around December December 27, 1995.12this document in 1995?12Q. And how could someone have accessed this document in 1995?14stored to download huge file which contains all the14A. Could unload this document as part of Linux15files related to Linux kernels.15source archive version by No. 1.3.52, after unloading16This file was numbered 1.3.42, and you could16the FTP site.17unpack the file and to fetch the file route.c from17Q. And do you know anyone who accessed this file20M. Yes. It was me; Linus Torvalds, who created the2021A. The question is more complicated. Yes, of2122Q. Okay.Q. Okay.23Torvalds. So2324But I can't many people who downloaded that24			1	
4available on?4that is in Exhibit 2?5A. November 16, 1994.5A. Yes.6Q. 1994?6Q. Okay. And was this document ever publicly7A. 5. I'm sorry. 1995.7available?8Q. So just to make the record clear, this version of8A. Yes. It was publicly available as part of kernel9route.c was publicly available on November 16, 1995?91.3.51.10A. Yes.10Q. And when was it first publicly available?11Q. And how and how could someone have accessed11A. On or around December December 27, 1995.12this document in 1995?12Q. And we was it first publicly available?13A. He could go to FTP site where Linux kernels are13in 1995?14stored to download huge file which contains all the14A. Could unload this document as part of Linux15files related to Linux kernels.15source archive version by No. 1.3.52, after unloading16This file was numbered 1.3.42, and you could16the FTP site.17unpack the file and to fetch the file route.c from17Q. And do you know anyone who accessed this file18there.18in 1995?19Q. Okay. Do you know anyone who accessed this file19A. Yes. It was me; Linus Torvalds, who created the20in 1995?20document, at least Alan Cox, to whom this patch was21A. The question is more complicated. Yes, of21submitted. But I downloaded it.			1	
5A. November 16, 1994.5A. Yes.6Q. 1994?6Q. Okay. And was this document ever publicly7A. 5. I'm sorry. 1995.7available?8Q. So just to make the record clear, this version of9route.c was publicly available on November 16, 1995?910A. Yes.10Q. And when was it first publicly available?11Q. And how and how could someone have accessed11A. On or around December December 27, 1995.12this document in 1995?12Q. And how could someone have accessed13A. He could go to FTP site where Linux kernels are13in 1995?14stored to download huge file which contains all the14A. Could unload this document as part of Linux15files related to Linux kernels.15source archive version by No. 1.3.52, after unloading16This file was numbered 1.3.42, and you could16the FTP site.17unpack the file and to fetch the file route.c from17Q. And do you know anyone who accessed this document18there.19A. Yes. It was me; Linus Torvalds, who created the20in 1995?20document, at least Alan Cox, to whom this patch was21A. The question is more complicated. Yes, of21submitted. But I downloaded it.22course. It was at least it was me, Alan Cox, Linus22Q. Okay.23Torvalds. So23(Exhibit No. 6 marked for24But I can't many people who downloaded that24 <td></td> <td></td> <td>1</td> <td></td>			1	
6Q. 1994?6Q. Okay. And was this document ever publicly7A. 5. I'm sorry. 1995.7available?8Q. So just to make the record clear, this version of9route.c was publicly available on November 16, 1995?910A. Yes.10Q. And how and how could someone have accessed10Q. And when was it first publicly available?11Q. And how and how could someone have accessed11A. On or around December December 27, 1995.12this document in 1995?12Q. And how could someone have accessed this document13A. He could go to FTP site where Linux kernels are13in 1995?14stored to download huge file which contains all the14A. Could unload this document as part of Linux15files related to Linux kernels.15source archive version by No. 1.3.52, after unloading16This file was numbered 1.3.42, and you could16the FTP site.17unpack the file and to fetch the file route.c from17Q. And do you know anyone who accessed this file19Q. Okay. Do you know anyone who accessed this file19A. Yes. It was me; Linus Torvalds, who created the20in 1995?20document, at least Alan Cox, to whom this patch was21A. The question is more complicated. Yes, of21Q. Okay.22Course. It was at least it was me, Alan Cox, Linus22Q. Okay.23Torvalds. So23(Exhibit No. 6 marked for24But I can't many people who downloaded that </td <td></td> <td></td> <td>1</td> <td></td>			1	
7A. 5. I'm sorry. 1995.7available?8Q. So just to make the record clear, this version of9route.c was publicly available on November 16, 1995?10A. Yes.10Q. And how and how could someone have accessed11Q. And how and how could someone have accessed10Q. And when was it first publicly available?12this document in 1995?12Q. And how could someone have accessed this document13A. He could go to FTP site where Linux kernels are13in 1995?14stored to download huge file which contains all the14A. Could unload this document as part of Linux15files related to Linux kernels.15source archive version by No. 1.3.52, after unloading16This file was numbered 1.3.42, and you could16the FTP site.17unpack the file and to fetch the file route.c from17Q. And do you know anyone who accessed this document18there.19A. Yes. It was me; Linus Torvalds, who created the20in 1995?20document, at least Alan Cox, to whom this patch was21A. The question is more complicated. Yes, of21submitted. But I downloaded it.22course. It was at least it was me, Alan Cox, Linus22Q. Okay.23Torvalds. So23(Exhibit No. 6 marked for24But I can't many people who downloaded that24			1	
8Q. So just to make the record clear, this version of 98A. Yes. It was publicly available as part of kernel9route.c was publicly available on November 16, 1995?91.3.51.10A. Yes.10Q. And when was it first publicly available?11Q. And how and how could someone have accessed11A. On or around December December 27, 1995.12this document in 1995?12Q. And how could someone have accessed this document13A. He could go to FTP site where Linux kernels are13in 1995?14stored to download huge file which contains all the14A. Could unload this document as part of Linux15files related to Linux kernels.15source archive version by No. 1.3.52, after unloading16This file was numbered 1.3.42, and you could16the FTP site.17unpack the file and to fetch the file route.c from17Q. And do you know anyone who accessed this document18there.18in 1995?19Q. Okay. Do you know anyone who accessed this file19A. Yes. It was me; Linus Torvalds, who created the20in 1995?20document, at least Alan Cox, to whom this patch was21A. The question is more complicated. Yes, of21submitted. But I downloaded it.22course. It was at least it was me, Alan Cox, Linus22Q. Okay.23Torvalds. So23(Exhibit No. 6 marked for24But I can't many people who downloaded that24identification.) <td></td> <td></td> <td></td> <td></td>				
9route.c was publicly available on November 16, 1995?91.3.51.10A. Yes.10Q. And when was it first publicly available?11Q. And how and how could someone have accessed11A. On or around December December 27, 1995.12this document in 1995?12Q. And how could someone have accessed this document13A. He could go to FTP site where Linux kernels are13in 1995?14stored to download huge file which contains all the14A. Could unload this document as part of Linux15files related to Linux kernels.15source archive version by No. 1.3.52, after unloading16This file was numbered 1.3.42, and you could16the FTP site.17unpack the file and to fetch the file route.c from17Q. And do you know anyone who accessed this document18there.18in 1995?19Q. Okay. Do you know anyone who accessed this file19A. Yes. It was me; Linus Torvalds, who created the20in 1995?20document, at least Alan Cox, to whom this patch was21A. The question is more complicated. Yes, of21submitted. But I downloaded it.22course. It was at least it was me, Alan Cox, Linus22Q. Okay.23Torvalds. So23(Exhibit No. 6 marked for24But I can't many people who downloaded that24identification.)				
10A. Yes.10Q. And when was it first publicly available?11Q. And how and how could someone have accessed11A. On or around December December 27, 1995.12this document in 1995?12Q. And how could someone have accessed this document13A. He could go to FTP site where Linux kernels are13in 1995?14stored to download huge file which contains all the14A. Could unload this document as part of Linux15files related to Linux kernels.15source archive version by No. 1.3.52, after unloading16This file was numbered 1.3.42, and you could16the FTP site.17unpack the file and to fetch the file route.c from17Q. And do you know anyone who accessed this document18there.18in 1995?19Q. Okay. Do you know anyone who accessed this file19A. Yes. It was me; Linus Torvalds, who created the20in 1995?20document, at least Alan Cox, to whom this patch was21A. The question is more complicated. Yes, of2122course. It was at least it was me, Alan Cox, Linus2223Torvalds. So23(Exhibit No. 6 marked for24But I can't many people who downloaded that24			1	
11Q. And how and how could someone have accessed11A. On or around December December 27, 1995.12this document in 1995?12Q. And how could someone have accessed this document13A. He could go to FTP site where Linux kernels are13in 1995?14stored to download huge file which contains all the14A. Could unload this document as part of Linux15files related to Linux kernels.15source archive version by No. 1.3.52, after unloading16This file was numbered 1.3.42, and you could16the FTP site.17unpack the file and to fetch the file route.c from17Q. And do you know anyone who accessed this document18there.18in 1995?19Q. Okay. Do you know anyone who accessed this file19A. Yes. It was me; Linus Torvalds, who created the20in 1995?20document, at least Alan Cox, to whom this patch was21A. The question is more complicated. Yes, of2122course. It was at least it was me, Alan Cox, Linus2223Torvalds. So23(Exhibit No. 6 marked for24But I can't many people who downloaded that24			1	
12this document in 1995?12Q. And how could someone have accessed this document13A. He could go to FTP site where Linux kernels are13in 1995?14stored to download huge file which contains all the14A. Could unload this document as part of Linux15files related to Linux kernels.15source archive version by No. 1.3.52, after unloading16This file was numbered 1.3.42, and you could16the FTP site.17unpack the file and to fetch the file route.c from17Q. And do you know anyone who accessed this document18there.18in 1995?19Q. Okay. Do you know anyone who accessed this file19A. Yes. It was me; Linus Torvalds, who created the20in 1995?20document, at least Alan Cox, to whom this patch was21A. The question is more complicated. Yes, of21submitted. But I downloaded it.23Torvalds. So23(Exhibit No. 6 marked for24But I can't many people who downloaded that24identification.)			1	· · · ·
13A. He could go to FTP site where Linux kernels are13in 1995?14stored to download huge file which contains all the14A. Could unload this document as part of Linux15files related to Linux kernels.15source archive version by No. 1.3.52, after unloading16This file was numbered 1.3.42, and you could16the FTP site.17unpack the file and to fetch the file route.c from17Q. And do you know anyone who accessed this document18there.18in 1995?19Q. Okay. Do you know anyone who accessed this file19A. Yes. It was me; Linus Torvalds, who created the20in 1995?20document, at least Alan Cox, to whom this patch was21A. The question is more complicated. Yes, of21submitted. But I downloaded it.23Torvalds. So23(Exhibit No. 6 marked for24But I can't many people who downloaded that24identification.)				
14stored to download huge file which contains all the14A. Could unload this document as part of Linux15files related to Linux kernels.15source archive version by No. 1.3.52, after unloading16This file was numbered 1.3.42, and you could16the FTP site.17unpack the file and to fetch the file route.c from17Q. And do you know anyone who accessed this document18there.18in 1995?19Q. Okay. Do you know anyone who accessed this file19A. Yes. It was me; Linus Torvalds, who created the20in 1995?20document, at least Alan Cox, to whom this patch was21A. The question is more complicated. Yes, of21submitted. But I downloaded it.23Torvalds. So23(Exhibit No. 6 marked for24But I can't many people who downloaded that24identification.)			1	-
15files related to Linux kernels.15source archive version by No. 1.3.52, after unloading16This file was numbered 1.3.42, and you could16the FTP site.17unpack the file and to fetch the file route.c from17Q. And do you know anyone who accessed this document18there.18in 1995?19Q. Okay. Do you know anyone who accessed this file19A. Yes. It was me; Linus Torvalds, who created the20in 1995?20document, at least Alan Cox, to whom this patch was21A. The question is more complicated. Yes, of21submitted. But I downloaded it.23Torvalds. So23(Exhibit No. 6 marked for24But I can't many people who downloaded that24identification.)		-	1	
16This file was numbered 1.3.42, and you could16the FTP site.17unpack the file and to fetch the file route.c from17Q. And do you know anyone who accessed this document18there.18in 1995?19Q. Okay. Do you know anyone who accessed this file19A. Yes. It was me; Linus Torvalds, who created the20in 1995?20document, at least Alan Cox, to whom this patch was21A. The question is more complicated. Yes, of21submitted. But I downloaded it.23Torvalds. So23(Exhibit No. 6 marked for24But I can't many people who downloaded that24identification.)		e	1	-
17unpack the file and to fetch the file route.c from17Q. And do you know anyone who accessed this document18there.18in 1995?19Q. Okay. Do you know anyone who accessed this file19A. Yes. It was me; Linus Torvalds, who created the20in 1995?20document, at least Alan Cox, to whom this patch was21A. The question is more complicated. Yes, of21submitted. But I downloaded it.22course. It was at least it was me, Alan Cox, Linus22Q. Okay.23Torvalds. So23(Exhibit No. 6 marked for24But I can't many people who downloaded that24identification.)				•
18there.18in 1995?19Q. Okay. Do you know anyone who accessed this file19A. Yes. It was me; Linus Torvalds, who created the20in 1995?20document, at least Alan Cox, to whom this patch was21A. The question is more complicated. Yes, of21submitted. But I downloaded it.22course. It was at least it was me, Alan Cox, Linus22Q. Okay.23Torvalds. So23(Exhibit No. 6 marked for24But I can't many people who downloaded that24identification.)			1	
19Q. Okay. Do you know anyone who accessed this file19A. Yes. It was me; Linus Torvalds, who created the20in 1995?20document, at least Alan Cox, to whom this patch was21A. The question is more complicated. Yes, of21submitted. But I downloaded it.22course. It was at least it was me, Alan Cox, Linus22Q. Okay.23Torvalds. So23(Exhibit No. 6 marked for24But I can't many people who downloaded that24			1	
20in 1995?20document, at least Alan Cox, to whom this patch was21A. The question is more complicated. Yes, of21submitted. But I downloaded it.22course. It was at least it was me, Alan Cox, Linus22Q. Okay.23Torvalds. So23(Exhibit No. 6 marked for24But I can't many people who downloaded that24identification.)			1	
21A. The question is more complicated. Yes, of21submitted. But I downloaded it.22course. It was at least it was me, Alan Cox, Linus22Q. Okay.23Torvalds. So23(Exhibit No. 6 marked for24But I can't many people who downloaded that24identification.)				
22course. It was at least it was me, Alan Cox, Linus22Q. Okay.23Torvalds. So23(Exhibit No. 6 marked for24But I can't many people who downloaded that24identification.)			1	
23Torvalds. So23(Exhibit No. 6 marked for24But I can't many people who downloaded that24identification.)			1	
24But I can't many people who downloaded that24identification.)			23	
			24	
	25	file, but I cannot tell who exactly it was.	25	Q. Handing you now what has been marked as

	Page 70		Page 72
1	Exhibit 6. It's using the numbers at the bottom	1	A. Yes.
2	center of the page, it starts with DEF00008567 and	2	Q. And what is it?
3	continues through DEF 00007601.	3	A. It's a so-called header file for routing cache.
4	Do you recognize this document?	4	It contains some definition comment shared by file
5	A. Yes.	5	route.c and another component of Linux kernel.
	Q. And what is this document?		-
6		6	Q. Okay. And so this is a true and correct copy of
7	A. It is the same document, slightly modified. Some	7	the route.h file from Linux version 2.0.1?
8	parts were fixed. Little features were added. And this	8	A. Yes.
9	document was published as part of Linux kernel version	9	Q. And was this document publicly available?
10	2.0.1.	10	A. Yes. This document was made publicly available
11	Q. Okay. And was this document publicly available?	11	simultaneous with file route.c.
12	A. Yes.	12	Q. Okay. And do you know if anyone accessed this
13	Q. And when was it first publicly	13	document in 19 July 1996?
14	A. This document was publicly available around	14	A. Yes. I did.
15	July 3, 1996.	15	Q. Okay. If you turn to the second page, at
16	Q. Okay. And how would someone have accessed this	16	line 65, and you see the word struct rtable?
17	document in 1996?	17	A. Yes.
18	A. Anyone could download Linux source archive	18	Q. What does that mean?
19	version 2.0.1, to unpack it, and to fetch file route.c	19	A. It's this structure contains all the data for
20	from there.	20	a particular routing cache entry. It contains all the
21	Q. Do you know if anyone accessed this document in	21	data, including destination where to send this packet
22	July 1996?	22	and linkage in hash table.
23	A. I downloaded the document.	23	Q. Okay. And you see line 67? It says struct
24	I don't know personally anyone who did the same,	24	rtable *rt next.
25	but I can surely say that this document, all the archive	25	A. Yes.
	Page 71		Page 73
1	was created and put to FTP site by Linus Torvalds. So	1	Q. What does that line do?
2	he obviously accessed the document.	2	A. This is pointer to the next element in hash
3	Q. Okay. Let's turn to the page ending in 8592.	3	table, which is in the linkage in routing hash table
4	A. Okay.	4	which we discussed before.
5	Q. And do you see line 1299? It says rt_cache_add?	5	Q. Right. So is it a pointer to the next element in
6	A. Yes.	6	the linked list?
		1	A. Yes.
7	Q. What is on line 1299?	7	
8	A. It's beginning of the first rt_cache-add.	8	Q. Okay. And if you look then on line 98, it says,
9	Q. We've discussed rt_cache-add earlier today.	9	extern struct rtable. That's where the line starts.
10	Is this the same function are there any	10	And then it says, ip_rt_hash_table.
11	differences between the rt_cache-add in this document	11	What does this line mean?
12	and the rt_cache-add we discussed in Exhibit 2?	12	A. This line declares describes hash table which
13	A. No. It is exactly the same function. The file	13	contains pointers to hash chains consisting of struct
14	was changed a little, but no changes were made for this	14	rtable, which we just discussed.
15	function.	15	Q. Okay. And so were records corresponding to
16	Q. Okay.	16	routing cache entries stored in that hash table?
17	(Exhibit No. 7 marked for	17	A. Yes.
18	identification.)	18	Q. Okay. All right. Mr. Kuznetsov, are any
19	Q. I am now handing you what has been marked as	19	defendants in this case paying you for your time in
20	Exhibit 7. Using the numbers at the bottom center of	20	connection with this deposition?
21	the page, it starts with DEF00008602, and it continues	21	A. No.
22	through DEF8605.	22	Q. Do you have as do you have any opinion on the
23	Do you see that?	23	validity of US patent No. 5,893,120?
24		24	

- validity of US patent No. 5,893,120?
- A. Yes. 24 25

24

25

A. Yes.

Q. Have you seen this document before?

MR. CASSADY: Object to form.

	Page 94		Page 96
1	Q. Okay. So the affidavit was written and	1	MR. ABSHER: Object to form.
2	controlled by Mr. Absher, right?	2	A. The code how it existed since 19 since 1997
3	THE INTERPRETER: (Russian).	3	and until 2008, when it was patched by Eric Dumazet,
4	A. Yes.	4	indeed does not collide with the patent. That is my
5	Q. And after reviewing it, you signed it, right?	5	statement. It doesn't collide with the patent.
6	A. Yes.	6	But the piece of code which was added by Eric
7	Q. But you didn't make any edits to the draft that	7	Dumazet indeed collide with the patent. But this code
8	Mr. Absher sent you, isn't that true?	8	is just an insignificant improvement of my code which I
9	A. If something was wrong there actually, the	9	wrote in 1985.
10	document contains only facts. If some facts were not	10	And then, due to its utter inefficiency, it was
11	true, I would edit it.	11	removed from the kernel at some point. For at least
12	But, actually, I did not have to do any changes.	12	five years, it wasn't used in Linux kernel. Not five.
13	All the facts are correct, to all I know.	13	More.
14	Q. So you signed the original version of the draft	14	But the idea remained the same, and we knew how
15	that Mr. Absher sent you?	15	to at some point, we knew how to improve it. So,
16	THE INTERPRETER: (Russian).	16	actually, it is seen from the code. The code which was
17	A. Yes.	17	added by Eric Dumazet is exactly the same place which
18	Q. Okay.	18	was used in my code dated about 1995, and does almost
19	A. What is original? I got a PDF file. I printed	19	exactly the same thing.
20	it. I signed it. I sent it to Alton Absher.	20	Q. Okay. Mr. Kuznetsov, let me ask you a very
21	Q. Okay. Now, Mr. Kuznetsov, we know you talked to	21	particular question.
22	some of the defendants in this case. We know you talked	22	So the code written by you in 1995 does not
23	to some of the attorneys for the defendants in this	23	actually collide
24	case.	24	A. No. My code
25	But have you ever attempted to contact Dr. Nemes,	25	Q. Sir, sir, Mr. Kuznetsov, let me finish my
	Page 95		Page 97
1	Page 95 who's the inventor on the patent in this case?	1	Page 97 question. One moment. I apologize. Let me finish my
1 2	who's the inventor on the patent in this case? MR. ABSHER: Object to form.	1 2	question. One moment. I apologize. Let me finish my question, okay?
	who's the inventor on the patent in this case?MR. ABSHER: Object to form.A. No.		question. One moment. I apologize. Let me finish my question, okay? It is true, sir, that the code written by you, in
2	who's the inventor on the patent in this case?MR. ABSHER: Object to form.A. No.Q. So you've never spoken with Dr. Nemes the	2	question. One moment. I apologize. Let me finish my question, okay?It is true, sir, that the code written by you, in 1995, does not actually collide with the patent in this
2 3 4 5	who's the inventor on the patent in this case?MR. ABSHER: Object to form.A. No.Q. So you've never spoken with Dr. Nemes the inventor on the patent in this case?	2 3 4 5	question. One moment. I apologize. Let me finish my question, okay?It is true, sir, that the code written by you, in 1995, does not actually collide with the patent in this case, because your code uses quite different techniques?
2 3 4 5 6	who's the inventor on the patent in this case?MR. ABSHER: Object to form.A. No.Q. So you've never spoken with Dr. Nemes the inventor on the patent in this case?MR. ABSHER: Object to form.	2 3 4 5 6	question. One moment. I apologize. Let me finish my question, okay?It is true, sir, that the code written by you, in 1995, does not actually collide with the patent in this case, because your code uses quite different techniques?Isn't that true, sir?
2 3 4 5 6 7	 who's the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. Q. So you've never spoken with Dr. Nemes the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. I know definitely, no. 	2 3 4 5 6 7	 question. One moment. I apologize. Let me finish my question, okay? It is true, sir, that the code written by you, in 1995, does not actually collide with the patent in this case, because your code uses quite different techniques? Isn't that true, sir? MR. ABSHER: Object to form.
2 3 4 5 6 7 8	 who's the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. Q. So you've never spoken with Dr. Nemes the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. I know definitely, no. Q. So have you ever spoken with any attorneys from 	2 3 4 5 6 7 8	 question. One moment. I apologize. Let me finish my question, okay? It is true, sir, that the code written by you, in 1995, does not actually collide with the patent in this case, because your code uses quite different techniques? Isn't that true, sir? MR. ABSHER: Object to form. A. No. It is not true. My code, written in 1995,
2 3 4 5 6 7 8 9	 who's the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. Q. So you've never spoken with Dr. Nemes the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. I know definitely, no. Q. So have you ever spoken with any attorneys from Bedrock about this case? 	2 3 4 5 6 7 8 9	 question. One moment. I apologize. Let me finish my question, okay? It is true, sir, that the code written by you, in 1995, does not actually collide with the patent in this case, because your code uses quite different techniques? Isn't that true, sir? MR. ABSHER: Object to form. A. No. It is not true. My code, written in 1995, directly collide with the patent. Actually, it used
2 3 4 5 6 7 8 9 10	 who's the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. Q. So you've never spoken with Dr. Nemes the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. I know definitely, no. Q. So have you ever spoken with any attorneys from Bedrock about this case? A. No. 	2 3 4 5 6 7 8 9 10	 question. One moment. I apologize. Let me finish my question, okay? It is true, sir, that the code written by you, in 1995, does not actually collide with the patent in this case, because your code uses quite different techniques? Isn't that true, sir? MR. ABSHER: Object to form. A. No. It is not true. My code, written in 1995, directly collide with the patent. Actually, it used exactly the same technique which was described in the
2 3 4 5 6 7 8 9 10 11	 who's the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. Q. So you've never spoken with Dr. Nemes the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. I know definitely, no. Q. So have you ever spoken with any attorneys from Bedrock about this case? A. No. Q. So is it fair to say that you've only heard the 	2 3 4 5 6 7 8 9 10 11	 question. One moment. I apologize. Let me finish my question, okay? It is true, sir, that the code written by you, in 1995, does not actually collide with the patent in this case, because your code uses quite different techniques? Isn't that true, sir? MR. ABSHER: Object to form. A. No. It is not true. My code, written in 1995, directly collide with the patent. Actually, it used exactly the same technique which was described in the patent up to letter.
2 3 4 5 6 7 8 9 10 11 12	 who's the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. Q. So you've never spoken with Dr. Nemes the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. I know definitely, no. Q. So have you ever spoken with any attorneys from Bedrock about this case? A. No. Q. So is it fair to say that you've only heard the story from the side of the defendants in this case? 	2 3 4 5 6 7 8 9 10 11 12	 question. One moment. I apologize. Let me finish my question, okay? It is true, sir, that the code written by you, in 1995, does not actually collide with the patent in this case, because your code uses quite different techniques? Isn't that true, sir? MR. ABSHER: Object to form. A. No. It is not true. My code, written in 1995, directly collide with the patent. Actually, it used exactly the same technique which was described in the patent up to letter. But this code was inefficient, and I removed that
2 3 4 5 6 7 8 9 10 11 12 13	 who's the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. Q. So you've never spoken with Dr. Nemes the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. I know definitely, no. Q. So have you ever spoken with any attorneys from Bedrock about this case? A. No. Q. So is it fair to say that you've only heard the story from the side of the defendants in this case? MR. ABSHER: Object to form. 	2 3 4 5 6 7 8 9 10 11 12 13	 question. One moment. I apologize. Let me finish my question, okay? It is true, sir, that the code written by you, in 1995, does not actually collide with the patent in this case, because your code uses quite different techniques? Isn't that true, sir? MR. ABSHER: Object to form. A. No. It is not true. My code, written in 1995, directly collide with the patent. Actually, it used exactly the same technique which was described in the patent up to letter. But this code was inefficient, and I removed that code. This code was not in Linux kernel for many years.
2 3 4 5 6 7 8 9 10 11 12 13 14	 who's the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. Q. So you've never spoken with Dr. Nemes the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. I know definitely, no. Q. So have you ever spoken with any attorneys from Bedrock about this case? A. No. Q. So is it fair to say that you've only heard the story from the side of the defendants in this case? MR. ABSHER: Object to form. A. Yes. First of all, yes, I heard the story only 	2 3 4 5 6 7 8 9 10 11 12 13 14	 question. One moment. I apologize. Let me finish my question, okay? It is true, sir, that the code written by you, in 1995, does not actually collide with the patent in this case, because your code uses quite different techniques? Isn't that true, sir? MR. ABSHER: Object to form. A. No. It is not true. My code, written in 1995, directly collide with the patent. Actually, it used exactly the same technique which was described in the patent up to letter. But this code was inefficient, and I removed that code. This code was not in Linux kernel for many years. And, at this time, Linux kernel didn't collide with the
2 3 4 5 6 7 8 9 10 11 12 13 14 15	 who's the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. Q. So you've never spoken with Dr. Nemes the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. I know definitely, no. Q. So have you ever spoken with any attorneys from Bedrock about this case? A. No. Q. So is it fair to say that you've only heard the story from the side of the defendants in this case? MR. ABSHER: Object to form. A. Yes. First of all, yes, I heard the story only from the only from defendants. 	2 3 4 5 6 7 8 9 10 11 12 13 14 15	 question. One moment. I apologize. Let me finish my question, okay? It is true, sir, that the code written by you, in 1995, does not actually collide with the patent in this case, because your code uses quite different techniques? Isn't that true, sir? MR. ABSHER: Object to form. A. No. It is not true. My code, written in 1995, directly collide with the patent. Actually, it used exactly the same technique which was described in the patent up to letter. But this code was inefficient, and I removed that code. This code was not in Linux kernel for many years. And, at this time, Linux kernel didn't collide with the patent because it used quite different expiration
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	 who's the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. Q. So you've never spoken with Dr. Nemes the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. I know definitely, no. Q. So have you ever spoken with any attorneys from Bedrock about this case? A. No. Q. So is it fair to say that you've only heard the story from the side of the defendants in this case? MR. ABSHER: Object to form. A. No. Q. So is it fair to say that you've only heard the story from the side of the defendants in this case? MR. ABSHER: Object to form. A. Yes. First of all, yes, I heard the story only from the only from defendants. But the patent is filed in US patent system. I 	2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	 question. One moment. I apologize. Let me finish my question, okay? It is true, sir, that the code written by you, in 1995, does not actually collide with the patent in this case, because your code uses quite different techniques? Isn't that true, sir? MR. ABSHER: Object to form. A. No. It is not true. My code, written in 1995, directly collide with the patent. Actually, it used exactly the same technique which was described in the patent up to letter. But this code was inefficient, and I removed that code. This code was not in Linux kernel for many years. And, at this time, Linux kernel didn't collide with the patent because it used quite different expiration technique.
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17	 who's the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. Q. So you've never spoken with Dr. Nemes the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. I know definitely, no. Q. So have you ever spoken with any attorneys from Bedrock about this case? A. No. Q. So is it fair to say that you've only heard the story from the side of the defendants in this case? MR. ABSHER: Object to form. A. Yes. First of all, yes, I heard the story only from the only from defendants. But the patent is filed in US patent system. I read it independently, not under guidance of anybody. 	2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17	 question. One moment. I apologize. Let me finish my question, okay? It is true, sir, that the code written by you, in 1995, does not actually collide with the patent in this case, because your code uses quite different techniques? Isn't that true, sir? MR. ABSHER: Object to form. A. No. It is not true. My code, written in 1995, directly collide with the patent. Actually, it used exactly the same technique which was described in the patent up to letter. But this code was inefficient, and I removed that code. This code was not in Linux kernel for many years. And, at this time, Linux kernel didn't collide with the patent because it used quite different expiration technique. Then, my idea from 1995, which existed in Linux
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18	 who's the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. Q. So you've never spoken with Dr. Nemes the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. I know definitely, no. Q. So have you ever spoken with any attorneys from Bedrock about this case? A. No. Q. So is it fair to say that you've only heard the story from the side of the defendants in this case? MR. ABSHER: Object to form. A. Yes. First of all, yes, I heard the story only from the only from defendants. But the patent is filed in US patent system. I read it independently, not under guidance of anybody. Q. Okay. So isn't it true, sir, that, as of 	2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18	 question. One moment. I apologize. Let me finish my question, okay? It is true, sir, that the code written by you, in 1995, does not actually collide with the patent in this case, because your code uses quite different techniques? Isn't that true, sir? MR. ABSHER: Object to form. A. No. It is not true. My code, written in 1995, directly collide with the patent. Actually, it used exactly the same technique which was described in the patent up to letter. But this code was inefficient, and I removed that code. This code was not in Linux kernel for many years. And, at this time, Linux kernel didn't collide with the patent because it used quite different expiration technique. Then, my idea from 1995, which existed in Linux kernel at least for two years, was returned in another
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19	 who's the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. Q. So you've never spoken with Dr. Nemes the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. I know definitely, no. Q. So have you ever spoken with any attorneys from Bedrock about this case? A. No. Q. So is it fair to say that you've only heard the story from the side of the defendants in this case? MR. ABSHER: Object to form. A. Yes. First of all, yes, I heard the story only from the only from defendants. But the patent is filed in US patent system. I read it independently, not under guidance of anybody. Q. Okay. So isn't it true, sir, that, as of December of 2010, you had reviewed the patent in this 	2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19	 question. One moment. I apologize. Let me finish my question, okay? It is true, sir, that the code written by you, in 1995, does not actually collide with the patent in this case, because your code uses quite different techniques? Isn't that true, sir? MR. ABSHER: Object to form. A. No. It is not true. My code, written in 1995, directly collide with the patent. Actually, it used exactly the same technique which was described in the patent up to letter. But this code was inefficient, and I removed that code. This code was not in Linux kernel for many years. And, at this time, Linux kernel didn't collide with the patent because it used quite different expiration technique. Then, my idea from 1995, which existed in Linux kernel at least for two years, was returned in another incarnation and applied to Linux kernel by Eric Dumazet.
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20	 who's the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. Q. So you've never spoken with Dr. Nemes the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. I know definitely, no. Q. So have you ever spoken with any attorneys from Bedrock about this case? A. No. Q. So is it fair to say that you've only heard the story from the side of the defendants in this case? MR. ABSHER: Object to form. A. Yes. First of all, yes, I heard the story only from the only from defendants. But the patent is filed in US patent system. I read it independently, not under guidance of anybody. Q. Okay. So isn't it true, sir, that, as of December of 2010, you had reviewed the patent in this case at least twice? 	2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20	 question. One moment. I apologize. Let me finish my question, okay? It is true, sir, that the code written by you, in 1995, does not actually collide with the patent in this case, because your code uses quite different techniques? Isn't that true, sir? MR. ABSHER: Object to form. A. No. It is not true. My code, written in 1995, directly collide with the patent. Actually, it used exactly the same technique which was described in the patent up to letter. But this code was inefficient, and I removed that code. This code was not in Linux kernel for many years. And, at this time, Linux kernel didn't collide with the patent because it used quite different expiration technique. Then, my idea from 1995, which existed in Linux kernel at least for two years, was returned in another incarnation and applied to Linux kernel by Eric Dumazet. At this point it started to collide again.
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21	 who's the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. Q. So you've never spoken with Dr. Nemes the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. I know definitely, no. Q. So have you ever spoken with any attorneys from Bedrock about this case? A. No. Q. So is it fair to say that you've only heard the story from the side of the defendants in this case? MR. ABSHER: Object to form. A. Yes. First of all, yes, I heard the story only from the only from defendants. But the patent is filed in US patent system. I read it independently, not under guidance of anybody. Q. Okay. So isn't it true, sir, that, as of December of 2010, you had reviewed the patent in this case at least twice? A. Yes. 	2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21	 question. One moment. I apologize. Let me finish my question, okay? It is true, sir, that the code written by you, in 1995, does not actually collide with the patent in this case, because your code uses quite different techniques? Isn't that true, sir? MR. ABSHER: Object to form. A. No. It is not true. My code, written in 1995, directly collide with the patent. Actually, it used exactly the same technique which was described in the patent up to letter. But this code was inefficient, and I removed that code. This code was not in Linux kernel for many years. And, at this time, Linux kernel didn't collide with the patent because it used quite different expiration technique. Then, my idea from 1995, which existed in Linux kernel at least for two years, was returned in another incarnation and applied to Linux kernel by Eric Dumazet. At this point it started to collide again. Q. Now
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22	 who's the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. Q. So you've never spoken with Dr. Nemes the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. I know definitely, no. Q. So have you ever spoken with any attorneys from Bedrock about this case? A. No. Q. So is it fair to say that you've only heard the story from the side of the defendants in this case? MR. ABSHER: Object to form. A. Yes. First of all, yes, I heard the story only from the only from defendants. But the patent is filed in US patent system. I read it independently, not under guidance of anybody. Q. Okay. So isn't it true, sir, that, as of December of 2010, you had reviewed the patent in this case at least twice? A. Yes. Q. And now, isn't it true, sir, that the code 	2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22	 question. One moment. I apologize. Let me finish my question, okay? It is true, sir, that the code written by you, in 1995, does not actually collide with the patent in this case, because your code uses quite different techniques? Isn't that true, sir? MR. ABSHER: Object to form. A. No. It is not true. My code, written in 1995, directly collide with the patent. Actually, it used exactly the same technique which was described in the patent up to letter. But this code was inefficient, and I removed that code. This code was not in Linux kernel for many years. And, at this time, Linux kernel didn't collide with the patent because it used quite different expiration technique. Then, my idea from 1995, which existed in Linux kernel at least for two years, was returned in another incarnation and applied to Linux kernel by Eric Dumazet. At this point it started to collide again. Q. Now THE VIDEOGRAPHER: Sorry. Just a minute,
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23	 who's the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. Q. So you've never spoken with Dr. Nemes the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. I know definitely, no. Q. So have you ever spoken with any attorneys from Bedrock about this case? A. No. Q. So is it fair to say that you've only heard the story from the side of the defendants in this case? MR. ABSHER: Object to form. A. Yes. First of all, yes, I heard the story only from the only from defendants. But the patent is filed in US patent system. I read it independently, not under guidance of anybody. Q. Okay. So isn't it true, sir, that, as of December of 2010, you had reviewed the patent in this case at least twice? A. Yes. Q. And now, isn't it true, sir, that the code written by you does not actually collide with the afore- 	2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23	 question. One moment. I apologize. Let me finish my question, okay? It is true, sir, that the code written by you, in 1995, does not actually collide with the patent in this case, because your code uses quite different techniques? Isn't that true, sir? MR. ABSHER: Object to form. A. No. It is not true. My code, written in 1995, directly collide with the patent. Actually, it used exactly the same technique which was described in the patent up to letter. But this code was inefficient, and I removed that code. This code was not in Linux kernel for many years. And, at this time, Linux kernel didn't collide with the patent because it used quite different expiration technique. Then, my idea from 1995, which existed in Linux kernel at least for two years, was returned in another incarnation and applied to Linux kernel by Eric Dumazet. At this point it started to collide again. Q. Now THE VIDEOGRAPHER: Sorry. Just a minute, Jason. I have four minutes of tape. Shall I
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22	 who's the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. Q. So you've never spoken with Dr. Nemes the inventor on the patent in this case? MR. ABSHER: Object to form. A. No. I know definitely, no. Q. So have you ever spoken with any attorneys from Bedrock about this case? A. No. Q. So is it fair to say that you've only heard the story from the side of the defendants in this case? MR. ABSHER: Object to form. A. Yes. First of all, yes, I heard the story only from the only from defendants. But the patent is filed in US patent system. I read it independently, not under guidance of anybody. Q. Okay. So isn't it true, sir, that, as of December of 2010, you had reviewed the patent in this case at least twice? A. Yes. Q. And now, isn't it true, sir, that the code 	2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22	 question. One moment. I apologize. Let me finish my question, okay? It is true, sir, that the code written by you, in 1995, does not actually collide with the patent in this case, because your code uses quite different techniques? Isn't that true, sir? MR. ABSHER: Object to form. A. No. It is not true. My code, written in 1995, directly collide with the patent. Actually, it used exactly the same technique which was described in the patent up to letter. But this code was inefficient, and I removed that code. This code was not in Linux kernel for many years. And, at this time, Linux kernel didn't collide with the patent because it used quite different expiration technique. Then, my idea from 1995, which existed in Linux kernel at least for two years, was returned in another incarnation and applied to Linux kernel by Eric Dumazet. At this point it started to collide again. Q. Now THE VIDEOGRAPHER: Sorry. Just a minute,